

UNIVERSIDADE DE PASSO FUNDO

Milene Giaretta

O PROGRAMA ESCOLA DE HACKERS E SUA
CONTRIBUIÇÃO PARA O DESENVOLVIMENTO
DO RACIOCÍNIO LÓGICO EM CRIANÇAS DO
ENSINO FUNDAMENTAL II

Passo Fundo

2018

Milene Giaretta

O PROGRAMA ESCOLA DE HACKERS E SUA
CONTRIBUIÇÃO PARA O DESENVOLVIMENTO
DO RACIOCÍNIO LÓGICO EM CRIANÇAS DO
ENSINO FUNDAMENTAL II

Dissertação apresentada ao Programa de Pós-Graduação em Ensino de Ciências e Matemática, do Instituto de Ciências Exatas e Geociências, da Universidade de Passo Fundo, como requisito parcial para a obtenção do grau de Mestre em Ensino de Ciências e Matemática, sob a orientação do Professor Dr. Adriano Canabarro Teixeira.

Passo Fundo

2018

CIP – Catalogação na Publicação

G435p Giaretta, Milene

O programa Escola de Hackers e sua contribuição para o desenvolvimento do raciocínio lógico em crianças do Ensino Fundamental II / Milene Giaretta. – 2018.

79 f. : il., color. ; 30 cm.

Orientador: Dr. Adriano Canabarro Teixeira.

Dissertação (Mestrado em Ensino de Ciências e Matemática) – Universidade de Passo Fundo, 2018.

1. Programação (Computadores) - Estudo e ensino. 2. Lógica no ensino. 3. Ensino Fundamental. I. Teixeira, Adriano Canabarro, orientador. II. Título.

CDU: 37:004

Catalogação: Bibliotecário Luís Diego Dias de S. da Silva – CRB 10/2241

Milene Giaretta

O PROGRAMA ESCOLA DE HACKERS E SUA
CONTRIBUIÇÃO PARA O DESENVOLVIMENTO
DO RACIOCÍNIO LÓGICO EM CRIANÇAS DO
ENSINO FUNDAMENTAL II

A banca examinadora, em 14 de dezembro de 2018, aprova a dissertação apresentada ao Programa de Pós-Graduação em Ensino de Ciências e Matemática – Mestrado Profissional da Universidade de Passo Fundo, como requisito parcial de exigência para obtenção de grau de Mestre em Ensino de Ciências e Matemática, na linha de pesquisa Tecnologias de informação, comunicação e interação aplicadas ao ensino de Ciências e Matemática.

Prof. Dr. Adriano Canabarro Teixeira - Orientador
Universidade de Passo Fundo

Prof. Dr. Adão Caron Cambraia
Instituto Federal Farroupilha

Prof. Dr. Marco Antonio Sandini Trentin
Universidade de Passo Fundo

AGRADECIMENTOS

Primeiramente agradeço a Deus por ter me iluminado e abençoado, me dado força e coragem para superar os momentos de dificuldades desta caminhada.

Em especial, agradeço ao professor orientador Adriano Canabarro Teixeira, pelas oportunidades, suporte, dedicação, paciência e disponibilidade para me auxiliar sempre que foi necessário na construção deste trabalho.

Agradeço imensamente a minha mãe Edilene que sempre acreditou em mim, dando forças para seguir em frente, além de nunca ter deixado de me apoiar e incentivar durante esta trajetória e ao meu irmão André, que mesmo longe sempre esteve na torcida, me incentivando e ajudando com o inglês.

Não poderia deixar de agradecer a oportunidade de vivenciar o Grupo de Pesquisa em Cultura Digital - GEPID, onde fiz novas amigas que levarei para a vida, aprendi, ensinei, encontrei suporte para superar as dificuldades, incentivei e fui incentivada. Experimentar este espaço onde ocorreram conversas, troca de conhecimentos, auxílio nas pesquisas e nas produções científicas foi essencial para a produção deste trabalho.

Não menos importante, também agradeço aos amigos e demais familiares por compreenderem a minha ausência em alguns momentos, incentivo e principalmente a tranquilidade que me passaram durante este período.

A todos, muito obrigada!

RESUMO

O presente trabalho pertence à linha de pesquisa Tecnologias de informação, comunicação e interação aplicadas ao ensino de Ciências e Matemática, do Programa de Pós-Graduação em Ensino de Ciências e Matemática. Seu objetivo é oportunizar um espaço diferenciado para o desenvolvimento de competências na área de programação de computadores e de raciocínio lógico, visto que a tecnologia tem revolucionado não só a forma como vivemos, mas também como aprendemos, tendo a disposição uma ampla quantidade de recursos digitais desenvolvidos especialmente para apoiar o processo de ensino-aprendizagem, visando aqueles educadores que desejam tornar as suas aulas mais interessantes, atrativas e interativas. Dentre os recursos disponíveis, destaca-se o ensino da programação, um importante elemento na formação do indivíduo, visto a possibilidade de aliar tecnologia e educação, aproximando-se do Programa Escola de Hackers, que busca proporcionar um espaço para desenvolvimento de conhecimentos de diversas áreas por meio da programação de computadores para estudantes do Ensino Fundamental. O projeto, já aplicado com resultados satisfatórios em relação ao desenvolvimento e aprimoramento das habilidades cognitivas dos participantes, é pautado em um Caderno Didático estruturado através de resolução de problemas e desafios, visando a utilização do *software* Scratch. Diante disso, esse trabalho propôs a adaptar o material didático a partir das perspectivas da Taxonomia de Bloom, visto que está voltada para a formação educacional e tem o objetivo de aprendizagem direcionada para a melhora no desenvolvimento cognitivo. Desse modo, o ensino não é centralizado na transferência de conteúdo e busca a ampliação expressiva dos resultados obtidos anteriormente, para então armazenar os dados em um blog, que foi desenvolvido na forma de material de apoio para professores. Neste sentido procura-se respostas para o seguinte questionamento: Quais as características e desdobramentos de um processo de ensino de programação de computadores para alunos de Ensino Fundamental para o desenvolvimento do raciocínio lógico? Nesse contexto cibernético, percebe-se que alfabetizar digitalmente os indivíduos, para que aprendam a transitar pelo mundo digital e a usar a tecnologia de maneira qualificada, é também função da escola. O produto educacional, que acompanha essa dissertação, encontra-se disponível em formato digital para acesso livre no portal Educapes, no seguinte endereço: <<http://educapes.capes.gov.br/handle/capes/431416>>.

Palavras chave: Escola de Hackers. Ensino. Programação de computadores. Raciocínio lógico.

ABSTRACT

The present work belongs to the line of research: Information technologies, communication and interaction applied to the teaching of Sciences and Mathematics, of the *Programa de Pós Graduação em Ensino de Ciências e Matemática* (Post-Graduate Program in Teaching Sciences and Mathematics). Its purpose is to provide a distinguished space for the development of computer programming and logical reasoning skills, since technology has revolutionized not only the way we live but also how we learn, with the provision of an ample amount of digital resources developed specifically to support the process of teaching learning, targeting those educators who wish to make their classes more interesting, attractive and interactive. Among the available resources, it is worth mentioning the teaching of programming, an important element in the formation of the individual, considering the possibility of allying technology and education, approaching the *Programa Escola de Hackers* (School of Hackers Program), which seeks to provide a space for the development of knowledge of diverse areas through the programming of computers for primary school students. The project, already applied with satisfactory results in relation to the development and improvement of the cognitive abilities of the participants, is based on a didactic book structured through problem solving and challenges, aiming at the usage of the Scratch software. Therefore, this work proposes to adapt the didactic material from Bloom Taxonomy perspectives, since it is geared towards educational formation and has the objective of learning directed to the improvement in cognitive development. Thus, teaching is not centralized in the transfer of content and seeks the expressive amplification of the results obtained previously, to then store the data in a blog, which will be developed in the form of teacher support material. In this sense, answers are sought for the following question: What are the characteristics and consequences of a teaching process of computer programming for elementary school students for the development of logical reasoning? In this cybernetic context, it is understood that to literate individuals digitally, so that they learn to transit through the digital world and to use technology in a qualified way, is also a function of the school. The educational product, which accompanies this dissertation, is available in digital format for free access on the Educapes portal at the following address: <<http://educapes.capes.gov.br/handle/capes/431416>>.

Keywords: Escola de Hackers. Teaching. Computer programming. Logical reasoning.

LISTA DE QUADROS

Quadro 1 - Sentenças utilizadas no teste com blocos lógicos.	32
Quadro 2 - Resultados gerados pelos três momentos de cada aluno.	38
Quadro 3 - Sequência didática pautada pelos níveis do domínio cognitivo de Bloom, com atividades do Caderno Didático.	53
Quadro 4 - Comparativo de resultados obtidos nos anos de 2017 e 2018 no software A Prova de Hackers pelos alunos vinculados ao Programa Escola de Hackers.	68
Quadro 5 - Comparativo de resultados obtidos nos anos de 2017 e 2018 através da plataforma Dr. Scratch pelos alunos vinculados ao Programa Escola de Hackers.	69

LISTA DE FIGURAS

Figura 1 - Blocos de comandos utilizados para a programação.	17
Figura 2 - Tela Inicial do Scratch: áreas, botões, abas e elementos gráficos.	20
Figura 3 - Modelo de questões.	34
Figura 4 - Visualização temporizada.	34
Figura 5 - Opções de resposta.	34
Figura 6 - Lista de respostas das questões.	34
Figura 7 - Gráfico de apresentação da melhoria na capacidade de retenção de informações dos alunos, representados por suas respectivas escolas, que são participantes do Programa Escola de Hackers.	35
Figura 8 - Elementos medidos pelo software Dr. Scratch.	37
Figura 9 - Gráfico de aproveitamento das escolas participantes do Programa Escola de Hackers.	38
Figura 10 - Representação dos seis níveis do domínio cognitivo.	51
Figura 11 - Representação dos seis níveis do domínio cognitivo, conforme estrutura do Caderno Didático.	57
Figura 12 - Elementos medidos pelo software Dr. Scratch para o projeto do grupo A, envolvendo as quatro operações básicas da Matemática.	62
Figura 13 - Elementos medidos pelo software Dr. Scratch para o projeto do grupo B, envolvendo a geometria básica da Matemática.	63
Figura 14 - Gráfico de apresentação da melhoria na capacidade de retenção de informações dos alunos, representados por suas respectivas escolas, que são participantes do Programa Escola de Hackers.	65
Figura 15 - Gráfico de aproveitamento das escolas participantes do Programa Escola de Hackers.	66

LISTA DE ABREVIATURAS E SIGLAS

CT - *Computational Thinking*

DATI - Departamento de Atenção da Terceira Idade

EMEF - Escola Municipal de Ensino Fundamental

EMEI - Escola Municipal de Educação Infantil

GEPID - Grupo de Pesquisa em Cultura Digital

IES – Instituição de Ensino Superior

MEC - Ministério da Educação

MIT - *Massachusetts Institute of Technology*

PCN - Parâmetros Curriculares Nacionais

PhD - *Philosophi Doctor*

PIBID - Programa Institucional de Bolsas de Iniciação à Docência

TI - Tecnologia da Informação

UPF - Universidade de Passo Fundo

URL - *Uniform Resource Locator*

SUMÁRIO

1	INTRODUÇÃO.....	10
2	TECNOLOGIA COMO FERRAMENTA DE ENSINO	14
2.1	Pensamento de Seymour Papert.....	14
2.2	Conhecendo o Scratch	17
2.3	Pensamento computacional.....	20
2.4	Programação de computadores e raciocínio lógico	24
2.5	Programa Escola de Hackers.....	27
<i>2.5.1</i>	<i>Gênese e dimensões do Programa.....</i>	<i>28</i>
<i>2.5.2</i>	<i>Organização metodológica do Programa Escola de Hackers</i>	<i>30</i>
<i>2.5.3</i>	<i>Resultados do projeto em 2017.....</i>	<i>31</i>
3	O ENSINO DE MATEMÁTICA PARA O DESENVOLVIMENTO DO RACIOCÍNIO LÓGICO.....	40
3.1	Ensino de Matemática segundo os PCN	40
3.2	Raciocínio lógico	46
3.3	Construção cognitiva pautada pela taxonomia de bloom	48
<i>3.3.1</i>	<i>Taxonomia dos objetivos educacionais</i>	<i>49</i>
<i>3.3.2</i>	<i>Utilização do Scratch no desenvolvimento de problemas matemáticos.....</i>	<i>52</i>
4	METODOLOGIA E O PRODUTO EDUCACIONAL.....	56
4.1	Aplicação e análise da sequência didática	58
4.2	Resultados da aplicação do Caderno Didático em 2018.....	64
<i>4.2.1</i>	<i>Comparativo dos resultados de 2017 e 2018</i>	<i>67</i>
5	CONSIDERAÇÕES FINAIS.....	70
	REFERÊNCIAS	72
	ANEXO A - Exposição do Blog	75

1 INTRODUÇÃO

Com o avanço tecnológico e a popularização das tecnologias da informação é comum presenciar adultos, idosos e crianças interagindo com e através de diversos recursos e equipamentos tecnológicos. Nessa perspectiva, surge uma tendência na educação: a inserção da programação de computadores nos currículos das escolas. Segundo Mitchel Resnick (2013), aprender a programar no século XXI é tão importante quanto aprender a ler, escrever e falar foi no século XX.

De acordo com Correia e Silva,

o uso das novas tecnologias não deve ser encarado como modismo ou com finalidade puramente comercial para aumentar o número de alunos, mas como ferramentas para a disseminação e construção do conhecimento. Uma das novas tecnologias que a escola tem procurado para estimular o aprendizado de seus alunos é o computador (2005, p. 12).

Desta forma, é possível observar a relação complementar estabelecida entre os termos educação e tecnologia. No entanto é preciso cautela e principalmente planejamento no desenvolvimento do ensino da programação para crianças. É necessário reconhecer que cada etapa da educação básica se configura de forma diferenciada, desse modo, é importante a adequação do ensino da programação a cada etapa, analisando o potencial e nível de conhecimento dos educandos, para então obter êxito com o uso desta ferramenta.

É válido ressaltar que a inserção do ensino de programação nos níveis da educação básica, prática que tem ganhado espaço nos últimos anos, deve ser articulada nas diversas áreas do conhecimento de forma interdisciplinar, priorizando o desenvolvimento pleno do educando. Esta nova ferramenta deve ser utilizada como instrumento que estimule a reflexão crítica, o raciocínio lógico, a desenvoltura na resolução de problemas simples e complexos, a interpretação do problema, o trabalho coletivo e a curiosidade. Neste contexto, Pólya nos traz que,

o problema pode ser modesto, mas se desafiar a curiosidade e puser em jogo as faculdades inventivas, quem o resolver pelos seus próprios meios experimentará o prazer e o triunfo da descoberta. Tais experiências, numa idade susceptível, poderão criar o gosto pelo trabalho mental e deixar, por toda a vida, uma marca indelével na mente e no carácter (2003, Prefácio).

Acredita-se que aprender a programar é de grande relevância, pois estamos inseridos em um mundo cada vez mais competitivo no qual a tecnologia é elemento fundamental de

desenvolvimento social. Desse modo, a introdução da programação na grade curricular das escolas terá um efeito substantivo no processo de construção do conhecimento. Seu potencial não se efetiva somente na área de Matemática, o ato de programar pode ser aplicado à arte, filosofia, história, geografia, física ou na química. Como pode é possível perceber, a programação de computadores pode enquadrar-se nas mais diversas áreas do desenvolvimento e também no aprendizado dos sujeitos, em especial quando é compreendida como processo e não fim.

Considerando que os jovens contemporâneos são bastante curiosos, vendo que hoje os recursos estão disponíveis em larga escala, a busca pela compreensão sobre como as coisas funcionam é um processo importante para programar computadores e construir conhecimentos. O desenvolvimento do raciocínio lógico é uma habilidade de grande importância para os jovens ampliarem a concentração, a percepção e a definição de hipóteses de resolução de problemas, levando-os ao desenvolvimento de estratégias eficazes nas soluções de problemas e tomadas de decisão consistentes, embasadas no conhecimento construído e no estudo dos problemas. Para Papert,

os cidadãos do futuro precisam lidar com desafios, enfrentar um problema inesperado para o qual não há uma explicação preestabelecida. Precisamos adquirir habilidades necessárias para participar da construção do novo ou então nos resignarmos a uma vida de dependência. A verdadeira habilidade competitiva é a habilidade de aprender. Não devemos aprender a dar respostas certas ou erradas, temos de aprender a solucionar problemas (2008, p. 122).

Durante este processo, o ato de programar computadores auxilia no desenvolvimento do pensamento crítico, da reflexão sobre determinado problema, favorecendo a análise de fatos e possibilidades, a fim de interpretá-los, uma das habilidades mais importantes na programação. Para se resolver um problema é necessária uma interpretação precisa dos dados e fatos ali postos, o que contribui no processo de compreender, explicar e declarar o verdadeiro sentido do enunciado. Além disso, a interação faz com que os jovens consigam se sentir estimulados pelo processo de aprendizagem, no qual a linguagem de programação e o resultado da execução são experienciados.

Para tanto, destaca-se que programação é um instrumento de ensino que perpassa todas as áreas do conhecimento, se utilizada de forma contextualizada pode nortear o desenvolvimento de diversas habilidades cognitivas no educando. Entre elas destacam-se: criatividade, curiosidade, resolução de problemas, criticidade, reflexão, interpretação, concentração, interação e raciocínio lógico, foco deste estudo.

A partir do que foi mencionado, define-se como problema de investigação o seguinte questionamento: Quais as características e desdobramentos de um processo de ensino de programação de computadores para alunos de Ensino Fundamental para o desenvolvimento do raciocínio lógico?

Diante disso, o **objetivo geral** do estudo foi aprimorar o caderno didático do programa Escola de Hackers a partir da Taxonomia de Bloom a fim de oportunizar um espaço diferenciado para o desenvolvimento de competências na área de programação de computadores e de raciocínio lógico. Os **objetivos específicos** do trabalho são: aprimorar uma metodologia de ensino de programação de computadores para estudantes do Ensino Fundamental II como alternativa para o desenvolvimento cognitivo; apresentar uma articulação teórica sobre Papert e Bloom, para subsidiar a construção do conhecimento.

Embora sejam inúmeras as iniciativas destinadas ao ensino de programação na educação formal, algumas merecem destaque pela forma como são organizadas e pela capilaridade de suas ações. Dentre elas, destaca-se o Programa Escola de Hackers realizado pela Prefeitura Municipal de Passo Fundo e que, desde 2014, já ensinou programação de computadores para mais de 1500 crianças e adolescentes da rede municipal de ensino. Marcado por metodologia própria que visa o desenvolvimento de competências, mais do que habilidade de programação por si só, tem sido campo de pesquisas na área de Educação e de Ensino.

Neste sentido, a consideração da Taxonomia de Bloom do domínio cognitivo para reorganização do Caderno Didático (produto educacional) foi estudada, tendo em vista que tal metodologia é estruturada em seis níveis de complexidade crescente, o que significa que, para adquirir uma nova competência que pertence ao próximo nível, o indivíduo precisa ter alcançado e dominado a habilidade do nível anterior, possuindo papel fundamental na organização e distribuição das atividades que abordam a programação de computadores, visando a qualidade que embasa o agir e o pensar dos estudantes ao atingir os resultados.

Diante desta inquietação, o primórdio deste estudo buscou embasamento em pesquisas nacionais, realizadas entre 2013 e 2016. Tais estudos tinham como propósito a reflexão sobre os efeitos da Taxonomia de Bloom aliada a abordagem da programação de computadores no ensino de Matemática. Nesse sentido, foram selecionados sete estudos para ilustrar a produção na área, tal aspecto ofereceu o suporte necessário para planejar e elaborar este trabalho. Para tanto, percebeu-se que a construção das aplicações de cursos, aulas ou seminários (sequência didática), avaliações e comparações encontradas nesses trabalhos, estão pautadas diretamente na Taxonomia de Bloom, mesmo utilizando-se de recursos tecnológicos,

mostrando que é possível entrelaçar a organização crescente dos objetivos educacionais com o ensino de programação de computadores, obtendo resultados significativos.

Como o ensino de programação ainda não faz parte do currículo escolar, foi preciso a aproximação ao Programa Escola de Hackers, um projeto inovador na área de programação de computadores, mantido pela Secretaria Municipal de Educação do município de Passo Fundo, que busca oportunizar um espaço diferenciado para o desenvolvimento de competências na área de programação de computadores e de raciocínio lógico. No ano de 2017, o programa aconteceu em dez escolas municipais e formou cerca de cinquenta jovens e no ano de 2018, o projeto ocorreu em cinco escolas municipais e graduou aproximadamente trinta adolescentes.

Em 2017 o Caderno Didático utilizado estava organizado com situações problema e desafios em ordem crescente de complexidade. Já no ano de 2018 o mesmo passou por uma reorganização à luz da Taxonomia de Bloom, visto que está direcionada para a formação educacional e tem como objetivo a aprendizagem desejando a melhora no desenvolvimento cognitivo, além de não ser centralizado na transferência de conteúdo, voltando a ser aplicado. Desta forma as análises dos resultados dos dois anos, foram realizadas de forma quantitativa com os mesmos instrumentos, sendo eles: um teste inicial com blocos lógicos, um *software* produzido dentro do Scratch e a evolução dos alunos identificada dentro da plataforma Dr. Scratch. A partir de tais procedimentos, os dados foram armazenados em um blog (Anexo A), que foi desenvolvido na forma de material de apoio, fornecendo fácil acesso e abrindo a possibilidade de contar com indicações, sugestões, comentários e críticas dos visitantes da plataforma.

Para que tais pressupostos sejam atendidos, o presente trabalho está organizado da seguinte forma: o Capítulo 2 trata das concepções de Seymour Papert, ligado ao desenvolvimento do pensamento computacional, através da programação de computadores por meio do *software* Scratch, ferramenta utilizada pelo Programa Escola de Hackers, que busca oportunizar espaço para aperfeiçoamento de conhecimentos de diversas áreas por meio da programação de computadores para estudantes do Ensino Fundamental. O Capítulo 3 trata do desenvolvimento do raciocínio lógico, tendo em vista a ludicidade no processo de ensino da Matemática através da programação de computadores. O Capítulo 4 trata da metodologia adotada para aprimoramento do Caderno Didático (produto educacional) e hospedagem de dados. E por fim, o Capítulo 5 nos traz as considerações finais deste trabalho até o momento.

2 TECNOLOGIA COMO FERRAMENTA DE ENSINO

O referencial teórico que subsidia este estudo está estruturado dentro do pensamento de Seymour Papert, precursor da inteligência artificial e defensor da inserção da programação de computadores no ensino, partindo de construções significativas por meio da tecnologia, para a aprendizagem como aborda o construcionismo.

Mentor do desenvolvimento do pensamento computacional, seus estudos serão valorizados e refletidos tendo em vista a programação de computadores, intermediada pela resolução de problemas. Além disso, o erro será visto como parte importante neste processo, vinculado diretamente à construção do próprio conhecimento, tendo como foco constante a ampliação do raciocínio lógico do indivíduo.

2.1 Pensamento de Seymour Papert

Segundo Papa (2017), Seymour Papert (1928-2016) foi um matemático e um dos pioneiros da inteligência artificial. Além disso, ele é reconhecido internacionalmente como o pensador seminal em relação aos computadores e a pedagogia para crianças. Um matemático por treinamento, sua colaboração com Jean Piaget na Universidade de Genebra levou-o a considerar o uso da Matemática para a compreensão do processo de aprendizagem e de pensamento. Papert foi proeminente educador estadunidense nascido em Pretória, na África do Sul, lecionava no *Massachusetts Institute of Technology* (MIT). Estudou na Universidade de Witwatersrand, graduado em 1949 e obteve um *Philosophiae Doctor* (PhD) em Matemática em 1952. Recebeu outro título de PhD, também em Matemática, na *Cambridge University* em 1959, onde foi orientado por Frank Smithies.

Papert obteve grande reconhecimento por aliar o uso de computadores e a educação. Foi o criador da linguagem de programação Logo (em 1967), inicialmente voltada para crianças, quando os computadores eram muitos limitados, não existia a interface gráfica e muito menos a internet. Defensor da utilização do computador como uma ferramenta educacional na aprendizagem, propôs o construcionismo, uma abordagem que defende que o sujeito constrói o seu próprio conhecimento por intermédio de alguma ferramenta, como é o caso do computador, por exemplo. Suas ideias se originaram de um conjunto de princípios de pensadores e tinham como objetivo um maior aproveitamento do uso das tecnologias no ensino, ou seja, seus estudos defendiam a construção do conhecimento baseada na realização

de uma ação concreta que resulta em um produto palpável, desenvolvido com o computador, partindo dos interesses de quem o produz, assim como Papert se posiciona,

o computador é a ferramenta que propicia à criança as condições de entrar em contato com algumas das mais profundas idéias em ciência, matemática e criação de modelos. [...] O computador deixa de ser o meio de transferir informação, e passa a ser a ferramenta com a qual a criança pode formalizar os seus conhecimentos intuitivos. O programa (a seqüência de ações ao computador) que a criança elabora é o espelho que reflete o seu conhecimento sobre um determinado assunto e o seu estilo de pensamento. Este programa, quando usado como objeto de reflexão, se torna uma poderosa fonte de aprendizagem (1985, p. 9-10).

O construcionismo, frequentemente se associa o adjetivo contextualizado, na perspectiva de destacar que tal produto, seja um texto, uma imagem, um mapa conceitual ou uma apresentação em slides, deve ter vínculo com a realidade da pessoa ou com o local onde será produzido e utilizado. Neste aspecto, Almeida destaca que,

o conhecimento não é fornecido ao aluno para que ele dê as respostas. É o aluno que coloca o conhecimento no computador e indica as operações que devem ser executadas para produzir as respostas desejadas. O programa fornece importantes pistas sobre o pensamento do aluno, uma vez que o pensamento está descrito explicitamente e a resposta do computador permite comparar o previsto com o obtido (2000, p. 33-34).

O construcionismo implica numa interação aluno-objeto (aluno-computador), mediada por uma linguagem de programação, como é o caso do Logo e o produto, resultado do processo, que pode ser exibido, visto, externalizado, discutido, examinado, admirado e analisado. Papert (2008) acredita que esta é a principal característica do construcionismo, pois ela permite a formação mental. Desta forma, o uso do computador é defendido como colaborador no processo de construção de conhecimentos, uma poderosa ferramenta educacional, que adapta os princípios do construtivismo cognitivo de Jean Piaget, a fim de melhor aproveitar o uso de tecnologias.

O termo Logo é uma referência a um termo grego que significa: pensamento, ciência, raciocínio lógico, cálculo, ou ainda, razão, linguagem, discurso e palavra. É também uma linguagem de programação voltada para o ambiente educacional. A linguagem é usada para comandar um cursor, normalmente representado por uma tartaruga, com o propósito de ensinar novos procedimentos além dos que ele já conhece, afim de criar desenhos ou programas. O grau de sofisticação desses desenhos ou programas pode variar desde um simples desenho de um quadrado, até a plotagem de um gráfico complexo, depende do nível do usuário, que pode ser tanto uma criança de 8 anos como um adulto.

Logo propõe uma metodologia de ensino que busca, através de uma linguagem semelhante a natural, facilitar a comunicação entre o usuário e o computador e proporcionar a criação de modelos através de formas geométricas e do raciocínio lógico. Propõe também, que o aluno seja construtor ativo de seus próprios conhecimentos, desenvolvendo assim sua capacidade intelectual.

O professor deve permitir a reflexão do aluno, ao contrário do modelo tradicional, assim como Papert se posiciona,

Nas salas de aula tradicionais, os professores tentam trabalhar em colaboração com a criança, mas, em geral, o material em si mesmo não gera espontaneamente problemas a serem pesquisados. [...] Uma das características importantes do trabalho com o computador é que o professor e aluno podem engajar-se numa verdadeira colaboração intelectual; juntos, podem tentar fazer com que o computador execute isto ou aquilo, e entender que ele realmente faz. Situações novas que nem o professor nem o aluno viram antes ocorrem frequentemente e assim o professor não tem que fingir que não sabe. Compartilhar o problema e a experiência de resolvê-lo permite a criança aprender com um adulto, não “fazendo o que o professor diz”, mas “fazendo o que o professor faz”. E uma das coisas que o professor faz é persistir num problema até que ele tenha sido completamente compreendido (1985, p. 143).

O aluno, através do erro, é levado a refletir sobre novas formas de resolução do problema, ou seja, tem a chance de aprender com seus próprios erros e é estimulado a tentar, pois quando acontece um equívoco, este torna-se um objeto de análise para que seja identificado e reformulado, desencadeando aprendizagem e desenvolvimento. Esse processo estabelece um ciclo de descrição-depuração, reflexão-depuração, que foi implantado na programação de computadores. Segundo Valente,

o processo de descrever, refletir e depurar não acontece simplesmente colocando o aluno em frente do computador. A interação aluno-computador precisa ser mediada por um profissional que conhece o Logo, tanto do ponto de vista computacional, quanto pedagógico e do psicológico. Esse é o papel do mediador no ambiente Logo (1998, p. 5).

Correia e Silva ainda nos trazem,

aqui, o erro deixa de ser uma arma de punição e passa a ser uma situação que nos leva a entender melhor nossas ações e conceitualização. É assim que a criança aprende uma série de conceitos antes de entrar na escola. Ela é livre para explorar e os erros são usados para depurar os conceitos e não para se tornarem a arma do professor (2005, p. 35).

Nesta linguagem, considera-se o erro como um importante fator de aprendizagem, o que oferece oportunidades para que o aluno entenda porque errou e busque uma nova solução

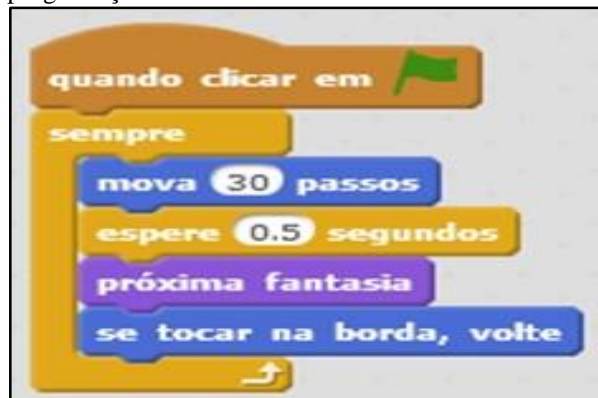
para o problema, investigando, explorando e descobrindo por si próprio, ou seja, a aprendizagem pela descoberta, assim como a teoria construcionista defende.

Aprender programação estimula a pensar de maneira mais criativa, desenvolve o raciocínio lógico, instiga a criar protótipos e a trabalhar em equipe. Para tanto, a ferramenta Scratch¹, constitui-se em uma linguagem de programação gráfica desenvolvida pelo MIT especialmente para crianças, oferecendo aos pequenos programadores, um ambiente de desenvolvimento acolhedor que permite criar animações, histórias interativas ou jogos em *browser*². Tal *software* permite a oportunidade de acesso à imaginação das crianças, pois são desafiados a colocar em prática o que idealizam, tendo à disposição uma interessante lista de operadores, controles, eventos, funções, palcos e personagens, cuja manipulação é simples.

2.2 Conhecendo o Scratch

O Scratch³ é um *software* livre, que foi inspirado nas linguagens Logo e Squeak, mas que pretende ser mais simples, fácil de utilizar e mais intuitiva. A linguagem foi feita para que as crianças criem programas sem a necessidade de digitar códigos complicados. Em vez disso, elas programam através de blocos de comandos que são encaixados uns nos outros (representados na Figura 1), formando um conjunto de instruções. Esses blocos facilitam o processo de descrição das instruções para a máquina (computador) uma vez que a sintaxe das indicações é definida pelo encaixe dos blocos, contribuindo para minimizar esse tipo de erro, que é muito comum em linguagem de programação baseada na codificação de comandos.

Figura 1 - Blocos de comandos utilizados para a programação.



Fonte: <https://scratch.mit.edu/>, adaptado pela autora.

¹ *Software* de programação livre constituído de uma versão atualizada da linguagem Logo, ou seja, utiliza-se de blocos de comandos para a programação e será abordado com mais ênfase na próxima seção.

² Qualquer navegador *web* de internet que permite o funcionamento online das produções criadas dentro do Scratch.

³ Disponível para acesso ao *software* livre em: <https://scratch.mit.edu/>.

O Scratch é um ambiente gráfico de programação que permite o desenvolvimento de aplicativos que integram recursos de multimídia de forma intuitiva. Tem por objetivo facilitar a introdução de conceitos de Matemática e de computação, bem como oportunizar o pensamento criativo, o raciocínio e o trabalho colaborativo. Segundo EDUScratch (1999), “a ferramenta Scratch foi concebida e desenvolvida como resposta ao problema do crescente distanciamento entre a evolução tecnológica no mundo e a fluência tecnológica dos cidadãos”. Nesta perspectiva Dullius diz que,

Como se percebe o Scratch pode ser bastante explorado na perspectiva da colaboração e da cooperação, já que ao pensar em um projeto faz-se necessário pensar no entendimento que o outro terá sobre o trabalho, bem como, as contribuições que poderão ser acrescentadas e ainda as dúvidas que serão levantadas, dentro de um processo de trocas constantes e estimulantes (2008, p. 31).

O termo “Scratch” provém da técnica de *scratching* utilizada pelos disco-jockeys do hip-hop, que giram os discos de vinil com as suas mãos para frente e para trás de modo a fazer misturas musicais originais. Com o Scratch, é possível fazer algo semelhante, misturando diferentes tipos de clipes de mídia de modos criativos, usando uma programação Matemática similar à feita nos programas de computador reais, porém de forma lúdica, simples e intuitiva.

Entre os recursos do Scratch destacam-se as competências para a resolução de problemas e para a concepção de projetos com raciocínio lógico, decomposição de problemas complexos em partes mais simples, identificação e eliminação de erros, desenvolvimento de ideias, desde a concepção até a concretização do projeto, concentração e perseverança, pois é voltado ao usuário infantil e jovem. Neste sentido, Papert coloca que,

é comum os estudantes falharem ao tentar resolver um problema, porque insistem em tentar resolvê-lo por inteiro de uma só vez; em muitos casos, seria muito mais fácil se reconhecessem que partes do problema podem ser resolvidas separadamente e depois unidas para lidar com o todo (2008, p. 90).

Oferece ainda uma linguagem de programação simples, em que é possível criar projetos que auxiliem na aprendizagem e no aperfeiçoamento de habilidades Matemáticas e computacionais, de modo a complementar e enriquecer o pensamento criativo e aprender a trabalhar de maneira colaborativa.

No site do Scratch é permitido criar uma conta para acessar, visualizar o interior e executar projetos já compartilhados por outros usuários, bem como compartilhar seus próprios projetos. Além disso, no site, no link “Ajuda”, você poderá encontrar tutoriais e vídeos que auxiliam na melhor compreensão e utilização do *software*. Para instalar o Scratch, versões 1.4

e 2.0, no computador, é possível utilizar o mesmo link. Se o usuário preferir, é possível realizar o projeto online, para isso basta clicar no link “Criar”. Diante de tais perspectivas, o ambiente de programação e de visualização do *software* Scratch 2.0 será apresentado, bem como noções básicas de palco e de atores, necessários para a construção de programas, jogos, histórias, etc.

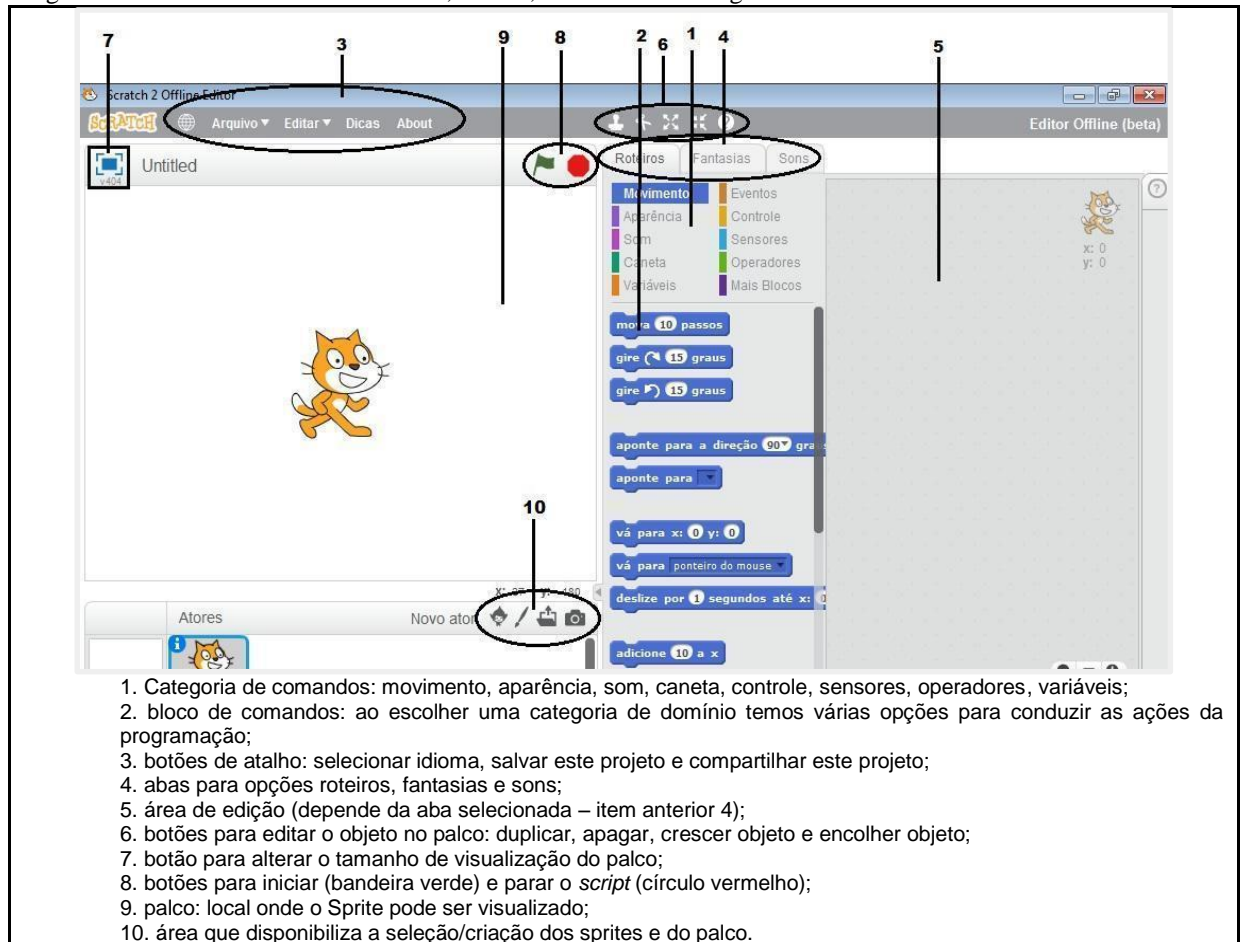
O Palco (Figura 2) é o lugar onde pode ser visualizada a execução de eventos ou projetos; é o ambiente no qual os objetos estão inseridos e é capaz mudar de aparência assumindo outros fundos de tela. Pode ser, por exemplo, um fundo com uma paisagem, um ambiente que represente um local, um período como o dia, a noite ou qualquer contexto no qual a história, ou seja, o jogo está inserido. As dimensões do palco são 480 unidades de largura e 360 unidades de altura e estão divididas entre as coordenadas X e Y. O centro do palco tem as coordenadas 0 em “X” e 0 em “Y”.

O desenvolvimento de qualquer aplicação é feito usando objetos, denominados sprites, que se constituem em imagens, desenhos, fotos ou qualquer outro objeto visual. A aparência de um sprite pode ser modificada pela sua apresentação em uma fantasia diferente. Qualquer imagem pode ser usada como uma fantasia; pode-se desenhar uma imagem no editor de pintura, importar de uma lista ou baixá-la de um website. Você pode dar instruções a um Sprite, mandar que ele se mova, que toque música, reaja a outros sprites, etc.

A lista de sprites contém miniaturas de todos os sprites do projeto, mostrando o nome, quantas fantasias e quantos *scripts* (roteiros, fantasias e sons vinculados aos objetos) cada um possui. Para ver e editar o *script* de um sprite é necessário clicar na miniatura na “Lista de Sprites” ou dar um duplo-clique no sprite na tela. Cada objeto utilizado na animação pode ter o seu comportamento programado a partir de um conjunto de blocos que se constituem em comandos que serão seguidos pelo personagem quando a animação for executada. Os comandos podem ser de movimento, por exemplo, indicando que o personagem vai se movimentar x passos em uma certa direção, que vai até uma posição específica. Outros comandos permitem modificar a aparência, a fantasia, cores e demais características dos objetos. Além destes, existem estruturas de controle que permitem executar atividades de forma repetida.

Desta forma, na Figura 2 aparece um objeto (sprite) com a imagem do gato que pode ser movimentada na tela a partir da utilização dos comandos desta linguagem de programação e indica os nomes dos campos, abas, botões da “Tela Inicial” para facilitar o uso deste aplicativo.

Figura 2 - Tela Inicial do Scratch: áreas, botões, abas e elementos gráficos.



Fonte: <https://scratch.mit.edu/>, adaptado pela autora.

É indiscutível que a programação de computadores é a nova habilidade fundamental de conhecimento a ser desenvolvida e esta tarefa deve ser propagada, preferencialmente, na escola, durante a Educação Básica, com intuito de garantir o acesso ao conhecimento e à cultura digital. Portanto, a inserção da programação é necessária para a evolução do pensamento computacional na educação básica, com o objetivo de fornecer recursos cognitivos que contribuam na resolução de problemas que perpassam todas as áreas do conhecimento, visando também o desenvolvimento do raciocínio lógico e outras habilidades cognitivas necessárias à aprendizagem de outros conteúdos.

2.3 Pensamento computacional

A tecnologia tem revolucionado não só a forma como vivemos, mas também como a que aprendemos. A quantidade de recursos digitais desenvolvidos especialmente para apoiar o processo de ensino-aprendizagem é ampla. Jogos eletrônicos, plataformas digitais, aplicativos

e *softwares* educacionais, entre outros, conformam um cardápio de opções variadas para aqueles educadores que desejam tornar as suas aulas mais interessantes, atrativas e interativas.

Nas palavras de Papert,

a questão central da mudança na Educação é a tensão entre a tecnicização e a não tecnicização, e aqui o professor ocupa a posição central. Desde a criação da máquina de imprimir não houve tão grande impulso no potencial para encorajar a aprendizagem tecnicizada. Há, porém, outro lado: paradoxalmente, a mesma tecnologia possui o potencial de destecnicizar a aprendizagem. Se isto ocorresse, eu contaria com uma mudança muito maior do que o surgimento, em cada carteira, de um computador programado para conduzir o estudante através de passos do mesmo velho currículo. Contudo, não é necessário sofismar sobre que mudança tem o maior alcance. O que é necessário é reconhecer que a grande questão no futuro da Educação é se a tecnologia fortalecerá ou subverterá a tecnicidade do que se tornou o modelo teórico e, numa grande extensão, a realidade da Escola (2008, p. 55).

A relação entre educação e tecnologia é antiga e foi se construindo a partir de momentos distintos. O surgimento do computador, por exemplo, significou uma mudança importante em termos de recursos educacionais. O que antes era feito no papel, passou a ser feito a partir de programas especializados na realização dessas tarefas. É necessário, portanto, incorporar os computadores na escola e difundir o seu uso em sala de aula.

A chegada da Internet também supôs uma “segunda” revolução nos processos de ensino-aprendizagem. O ambiente virtual, além de trazer consigo um intenso fluxo de comunicação e informação, também simplificou e facilitou uma série de atividades do nosso cotidiano através de aplicativos interativos.

A ideia do pensamento computacional com visão para a educação do século XXI, surgiu com Jeannette Wing, que o define como um conjunto de competências e habilidades básicas que os alunos devem desenvolver a partir do Ensino Básico. Tal autora ainda coloca que:

pensamento computacional é uma forma para seres humanos resolverem problemas; não é tentar fazer com que seres humanos pensem como computadores. Computadores são tediosos e enfadonhos; humanos são espertos e imaginativos (2006, p. 35).

Não se trata de saber navegar na internet, enviar e-mails ou usar de ferramentas computacionais como o processador de texto ou planilha de cálculo. Pensamento computacional é saber usar o computador como um instrumento de ampliação do poder cognitivo e operacional humano, em outras palavras, é usar computadores e redes para aumentar a produtividade, inventividade e criatividade. Trata-se da ativação de incumbências

cognitivas que possam ser realizadas pelo computador e, neste caso, saber programá-lo para cumprir tais encargos. Jeannette Wing ainda apresenta que “pensamento computacional é uma habilidade fundamental para todos, não somente para cientistas da computação” (2006, p. 33).

A popularização da computação, somada ao crescente uso da internet, mudou completamente a forma das pessoas se relacionarem com as informações. Enquanto antes havia a dependência dos jornais impressos, rádio ou televisão para que se os cidadãos se mantivessem informados, atualmente, a informação pode ser obtida a partir de uma infinidade de dispositivos conectados à internet, tais como celulares, tablets, notebooks, relógios digitais e outros. Com isto, mudou completamente a forma e a velocidade com que as pessoas têm acesso às informações que desejam.

Diante disso, percebe-se a necessidade de desenvolver o pensamento computacional (em inglês: *computational thinking* ou CT), que é uma expressão cunhada por Seymour Papert em 1980 e novamente em 1996. Tornando-se popular devido a um artigo intitulado *Computational Thinking* de Jeannette M. Wing (2006) e tem gerado um grande número de iniciativas de uso de tecnologia nas escolas, em geral envolvendo atividades de programação e robótica.

Papert define que pensamento computacional é um processo de resolução de problemas que inclui determinadas características de disposições. Este processo, por sua vez, é essencial para o desenvolvimento de programas para computador, mas pode também ser utilizado como suporte para a resolução de problemas, independentemente da disciplina, incluindo Matemática, Ciências e as áreas humanas. Dentro desse contexto, Jeannette Wing destaca que,

pensamento computacional envolve a resolução de problemas, projeção de sistemas, e compreensão do comportamento humano, através da extração de conceitos fundamentais da ciência da computação. O pensamento computacional inclui uma série de ferramentas mentais que refletem a vastidão do campo da ciência da computação. [...] Ao resolver um problema eficientemente, podemos questionar se uma solução aproximada é boa o suficiente e se falsos positivos ou falsos negativos são permitidos. O pensamento computacional é reformular um problema aparentemente difícil em um problema que sabemos como resolver, talvez por redução, incorporação, transformação ou simulação (2006, p. 33).

A ideia de que a programação de computadores ajuda a pensar melhor não é nova. Papert já mencionava a importância dessa atividade para o processo de construção de conhecimento e para o desenvolvimento do pensamento. Em 1971, ele argumentou que a computação pode ter “um impacto profundo por concretizar e elucidar muitos conceitos anteriormente sutis em psicologia, linguística, biologia, e os fundamentos da lógica e da

Matemática” (PAPERT, 1971, p. 2). Tais aspectos são possíveis pelo fato de proporcionar a uma criança a capacidade “de articular o trabalho de sua própria mente e, particularmente, a interação entre ela e a realidade no decurso da aprendizagem e do pensamento” (PAPERT, 1971, p. 3).

Nesse contexto cibernético, alfabetizar digitalmente os indivíduos para que aprendam a transitar pelo mundo digital e a usar a tecnologia de maneira qualificada, é também função da escola. Para Rushkoff,

quando nós, os humanos adquirimos linguagem, aprendemos não somente a ouvir, mas a falar. Quando ganhamos a escrita, nós aprendemos não apenas a ler, mas a escrever. E na medida em que nos movemos em direção a uma realidade crescente digital, nós precisamos aprender não apenas a usar programas, mas a fazê-los também. No panorama emergente, altamente programado, ou você criará o *software* ou será o *software*. Simples assim: programe ou será programado (2012, p. 7).

Se observarmos ao nosso redor, o mundo em que vivemos é, cada vez mais, mediado pela tecnologia. A cada minuto um novo aplicativo ou plataforma digital é criada para facilitar o nosso dia a dia, ou mesmo propor uma solução para algum problema social a partir de uma abordagem colaborativa e com uma velocidade nunca antes vista. Hoje, para que um indivíduo seja autônomo, capaz de transformar ativamente a sua realidade, não basta saber usar determinado aplicativo ou ferramenta digital: é preciso entender e “falar” essa nova linguagem que dá corpo e permite o desenvolvimento tecnológico constante e veloz no qual o mundo está inserido. Essa nova linguagem é a programação. Desta forma, a escola deve atuar para facilitar a compreensão dessa nova linguagem, apoiando o desenvolvimento da capacidade criadora do aluno, colocando-o como produtor de novas tecnologias. Para a concretização de tal sistema é preciso introduzir progressivamente a aprendizagem da lógica e da linguagem de programação no currículo das escolas.

Nesse sentido, embora algumas disciplinas como língua portuguesa, estimulem a capacidade do jovem como criador de conteúdo, em outras, como é o caso de Matemática, a tecnologia aparece como uma facilitadora do processo de aprendizagem. Hoje, um professor pode tanto trabalhar um conteúdo de geometria, pedindo que seus alunos construam gráficos e desenhos geométricos através das ferramentas digitais disponíveis, como também propondo que os jovens criem um aplicativo ou plataforma digital que possibilite a criação desses gráficos e desenhos.

Uma outra maneira de potencializar o ensino de programação nas escolas é incentivar professores e estudantes a criar aplicativos para a resolução de problemas sociais que

envolvam a comunidade onde vivem. São ideias simples e inovadoras de como abordar o uso da tecnologia no processo de ensino-aprendizagem de uma forma mais dinâmica e interativa, que poderiam ser incorporados na atual proposta de Base Nacional Comum Curricular.

Com as mudanças tecnológicas e informacionais que a sociedade sofreu nos últimos anos, é cada vez mais necessário que as instituições de ensino incorporem à sua grade consolidada novas matérias que atendam as demandas do século XXI, sendo o ensino de programação nas escolas uma das mais importantes.

2.4 Programação de computadores e raciocínio lógico

Aprender a programar fornece vários benefícios, tanto para os estudantes como para as escolas que lecionam essa habilidade. Ao programar, o aluno lida com vários desafios mentais, que o forçam a pensar de forma abstrata, sequencial e organizada. Assim como jogar xadrez ou estudar um instrumento musical cria novos padrões de conexões neuronais, programar também estimula partes do cérebro relacionadas à lógica e ao pensamento matemático. Dessa forma, a programação nas escolas é uma ótima maneira, não só para os alunos se tornarem bons programadores, mas também para desenvolverem suas capacidades em outras matérias de exatas.

A programação proporciona ao aluno um universo desafiador que o desperta para pensar. Isso implica em um sistema de aprendizagem ativa, onde a composição do conhecimento se dá a partir das ações físicas ou mentais do aluno. Programar abrange uma série de capacidades, das quais destacam-se a criatividade, capacidade de resolução de problemas, trabalho em grupo e o raciocínio lógico.

A criatividade pode ser assimilada como um modo de utilizar e reutilizar a informação, mas também pode ser uma ação. Sobre a criatividade, Vygotsky reflete que “entre as questões mais importantes da psicologia infantil e da pedagogia, encontra-se a da capacidade criadora das crianças, a da promoção desta capacidade e a da sua importância no desenvolvimento geral e maturação da criança” (1989, p. 13).

Neste sentido, o pensamento criativo apresenta duas características principais: a demonstração da autonomia sobre o que fazer e como fazer e a orientação para criação de novas ideias, partindo daquelas já existentes. Uma das capacidades usadas quando se está programando, fundamental para que haja a interação com o computador, o objeto a ser gerado e a pessoa que está elaborando, é o uso da imaginação, ou seja, servir-se da capacidade criadora.

Além da criatividade, a capacidade de resolução de problemas é uma habilidade Matemática que leva a desenvolver inúmeras outras competências. Ponte et al. (2007) consideram que a resolução de problemas pode ser uma importante contribuição na formação integral do indivíduo. Pólya (2003), ainda, define a resolução de problemas como uma arte que todos podem aprender, o objetivo é desenvolver a capacidade de pensar. Para o autor, a resolução de problemas inclui quatro etapas: compreensão do problema, elaboração de um plano, execução do plano e verificação dos resultados.

Além do desenvolvimento da criatividade e da habilidade na resolução de problemas, o trabalho em grupo também se faz importante para a aprendizagem eficiente através de programação. A necessidade de pensar soluções para um determinado problema e, posteriormente, expor a um grupo que deseja também resolvê-lo a fim de escolher aquela que melhor responde à natureza do desafio, representa um poderoso contexto de aprendizagem partilhada, onde as diferentes hipóteses podem ser questionadas e aprimoradas pelo grupo. Além disso, nesta dinâmica de trocas estabelecida em torno do ato de pensar uma solução para determinada situação, os alunos que apresentam mais dificuldades se beneficiam das competências de seus colegas que, ao ajudar o grupo, acabam por revisar e reformular constantemente aquilo que já sabem, assim como enfoca o Programa de Matemática do Ensino Básico,

os alunos devem ser capazes de, oralmente e por escrito, descrever a sua compreensão matemática e os procedimentos matemáticos que utilizam. Devem, igualmente, explicar o seu raciocínio, bem como interpretar e analisar a informação que lhes é transmitida por diversos meios. Estas capacidades desenvolvem-se comunicando por uma variedade de formas e aperfeiçoando os seus processos de comunicação (BRASIL, 2007, p. 5).

Segundo Azerêdo (2014), o uso de problemas no Ensino Fundamental desenvolve no aluno a capacidade de expressar ideias e opiniões, constrói o intelectual, instiga no levantamento de hipóteses e edifica a confiança em traçar estratégias. O aluno, ao ser instigado, irá criar métodos para as resoluções dos problemas.

A criatividade na elaboração de estratégias é importante. Muitos alunos perdem tempo tentando solucionar a estratégia adequada para resolver o problema, dentre aquelas ensinadas pelo professor. Outros já se permitem criar, aparecendo com soluções surpreendentes (BRASIL, 2008, p. 50).

Para resolver situações problemas, os alunos usam os seus conhecimentos prévios: tirar dados, escrever algoritmos, contar, comparar, etc e aplicam aos problemas e a partir daí

obtem um novo conhecimento. Porém, é interessante que o professor faça uma mediação questionando os alunos quais outros meios eles poderiam utilizar para conseguir um resultado, assim como ressalta Dante,

devemos propor aos estudantes várias estratégias de resolução de problemas, mostrando-lhes que não existe uma única estratégia, ideal e infalível. Cada problema exige uma determinada estratégia. A resolução de problemas não deve se constituir em experiências repetitivas, através da aplicação dos mesmos problemas (com outros números) resolvidos pelas mesmas estratégias. O interessante é resolver diferentes problemas com uma mesma estratégia e aplicar diferentes estratégias para resolver um mesmo problema. Isso facilitará a ação futura dos alunos diante de um problema novo (2000, p. 59-60).

A importância da interação entre aluno e professor também é importante e significativa. Vygotsky (1989) ressalta a importância do inter-relacionamento entre os membros de uma escola para a ampliação das possibilidades de sucesso na aprendizagem enquanto processo de interação. As ações educativas devem ocorrer num ambiente de interação social, promovendo o intercâmbio entre os diferentes sujeitos escolares. A importância do trabalho em grupo vai além do fato de o aluno trabalhar a sua aprendizagem ao mesmo tempo que seu colega, pois passa pelo aumento do índice de responsabilidade e de autoestima dos estudantes, promovendo, assim, o relacionamento entre alunos e as suas capacidades interpessoais.

Além destas habilidades, a capacidade desenvolvida no ato de programar computadores impulsiona a aprendizagem eficiente e também o raciocínio lógico. O pensamento lógico parte do indivíduo e da construção de relações existentes entre os objetos, sendo elemento fundamental para que a relação sujeito e objeto tome sentido, ocorrendo uma crescente capacidade do intelecto humano em alavancar o pensamento lógico e organizar a própria lógica das atividades. Notadamente, não se trata de um conceito ou de uma capacidade que se possa ensinar em um processo direto de relação educativa entre professor e aluno, por exemplo.

O raciocínio lógico, fundamental ao processo de aprendizagem em qualquer área do conhecimento, se desenvolve nos momentos em que os indivíduos se deparam com situações em que precisam perceber a demanda e estruturar o pensamento, detalhadamente, para resolver determinado problema. Além disso, tal habilidade também se torna perceptível quando os estudantes precisam criar estratégias e testá-las para aferir a eficiência da solução e, posteriormente, executá-las para identificar sua eficácia. Desta forma, Rampanelli et al. (2015, p. 236) ressaltam que “é possível intuir que o raciocínio lógico não é fruto de atos

espontâneos dos indivíduos, mas de atividades que os envolvam, desafiem e que possibilitem a criação de uma gama potencialmente infinita de soluções”.

Enfim, considerando as características de criatividade, capacidade de resolução de problemas, trabalho em grupo e o raciocínio lógico desenvolvidas em processos de programação de computadores, acredita-se fortemente, que o ato de programar computadores, por muito tempo associado a profissionais da área de informática, pode constituir uma alternativa poderosa de qualificação da educação, subvertendo a aplicação interdisciplinar dos recursos tecnológicos digitais no ensino, conforme elencado nos PCN do Ensino Fundamental.

2.5 Programa Escola de Hackers

É indiscutível a importância que a área de Tecnologia da Informação (TI) tem assumido no mundo contemporâneo, instituindo-se como elemento determinante de desenvolvimento de pessoas, empresas, municípios, estados e países. Motor do progresso econômico e vetor de agregação social, as tecnologias modernas transformam profundamente o mundo e abrem inúmeras perspectivas de ampliação da qualidade de vida dos indivíduos, de transparência para as instituições públicas e de mudanças da sociedade como um todo. Entretanto, é possível identificar claramente dois paradoxos. Primeiro, apesar da importância fundamental da área e de seu crescimento, existe um déficit de profissionais. Segundo, apesar do potencial democrático das tecnologias em diminuir a distância existente entre as diferentes camadas sociais, bem como para a qualificação da educação, o Brasil continua a ocupar as últimas posições nos rankings mundiais de qualidade em educação⁴.

Passo Fundo ocupa lugar estratégico e de destaque no cenário da Tecnologia de Informação fora de capitais, uma vez que conta com uma grande base de empresas na área, associações de profissionais, incubadoras empresariais e parques científicos e tecnológicos, além de vários cursos de nível médio e superior de graduação e pós-graduação. Assim, é fundamental que se criem espaços e oportunidades para que se desenvolva competência nesta área já no Ensino Fundamental.

O Programa Escola de Hackers, inovador na área de programação de computadores, é um desses espaços promovidos pela Secretaria Municipal de Educação do município de Passo

⁴ Acesso ao ranking mundial de qualidade da educação, conforme a Associação Brasileira de Educação, disponível em: <<https://goo.gl/jiNpjz>>.

Fundo, que atende 41 escolas de Ensino Fundamental, nas zonas urbana e rural, conforme exposto no site oficial⁵ da prefeitura municipal.

2.5.1 Gênese e dimensões do Programa

A idealização do Programa veio de diversas ações que aconteciam na Universidade de Passo Fundo (UPF) em 2013, mais especificamente no Grupo de Pesquisa em Cultura Digital (GEPID) e no projeto de extensão Mutirão pela Inclusão Digital. A ação responsável por esta promoção foi a Olimpíada de Programação de Computadores para Estudantes de Ensino Fundamental, que já exigia a utilização do *software* Scratch pelas equipes participantes. No final de 2013, o programa foi sistematizado por professores colaboradores do grupo e do projeto de extensão anteriormente citado, cujo principal objetivo é oportunizar espaço para desenvolvimento de competências na área de programação de computadores e de raciocínio lógico para estudantes do Ensino Fundamental da rede pública de ensino do município de Passo Fundo.

A partir disso, no início de 2014, iniciaram-se as atividades da Escola de Hackers, com a inscrição de 21 escolas e de 312 alunos matriculados. Em cada escola, foram realizadas oficinas semanais de aproximadamente duas horas, por um monitor, acadêmicos bolsistas das Instituições de Ensino Superior (IES) envolvidas⁶ e alunos vinculados ao Programa Institucional de Bolsas de Iniciação à Docência (PIBID) da Licenciatura em Matemática da UPF, usando o *software* Scratch.

Ao longo do ano as atividades trabalhadas nas oficinas foram elaboradas e discutidas em reuniões semanais com os monitores e a equipe organizadora do programa. Elas abordavam conceitos iniciais de programação e raciocínio lógico, noções de variável, de listas, operadores, sensores, controle, entre outros.

Com a base de programação construída, os alunos passaram a adquirir autonomia para a criação de seus projetos. Dessa forma, as atividades passaram a explorar esta característica, através de animações e jogos. Inicialmente, os monitores apresentaram aos seus alunos os conceitos de jogo e de animação. Na sequência, eles os desafiaram a elaborar um jogo e uma história animada. Os alunos, então divididos em grupos e motivados, realizaram os dois desafios.

⁵ Disponível para acesso em: <http://www.pmpf.rs.gov.br/>.

⁶ IES envolvidas com o Programa Escola de Hackers: Universidade de Passo Fundo (UPF), Faculdade Meridional (IMED) - Campus Passo Fundo e Instituto Federal do Rio Grande do Sul (IFRS) - Campus Passo Fundo.

No final do ano de 2014 foi realizada a apresentação dos projetos desenvolvidos pelos alunos na sua comunidade escolar. Dessa forma, cada escola organizou um espaço e marcou uma data para a execução de tal atividade. No dia escolhido pela escola, cada grupo de alunos descreveu as concepções para elaboração de seus projetos e a sua finalização. Foram dois os desafios lançados para os alunos realizarem seus projetos: a construção de um jogo e a animação de uma história natalina. Participaram das apresentações dos alunos representantes da UPF, direção, coordenador do laboratório de informática e alunos de cada escola envolvida.

Após a finalização deste programa piloto, o desdobramento foi em 2015, e contava com a continuidade das oficinas nas escolas para novos estudantes, com o auxílio daqueles que já participaram do programa, e com a realização de oficinas avançadas de programação, na UPF, para estudantes destaques desta primeira experiência, envolvendo a robótica educacional.

Após o programa não ocorrer no ano de 2016, retomou suas atividades em 2017, e foram quatro frentes diferentes: Berçário de Hackers⁷, Escola de Hackers⁸, Escola de Hackers Avançada⁹ e Academia White Hat¹⁰.

A *Escola de Hackers*, foco deste estudo, realizou suas atividades semanalmente em algumas escolas e quinzenalmente em outras, sendo que as aulas semanais possuíam duração de 2h e as quinzenais, duração de 4h. O grupo de trabalho foi composto por acadêmicos dos cursos de Psicologia, Análise e Desenvolvimento de Sistemas, Ciência da Computação e Programa de Pós-Graduação em Ensino de Ciências e Matemática, além de professores da área de Psicologia e Computação da Universidade de Passo Fundo.

Neste ano de 2018, o programa conta com a continuidade das oficinas nas quatro frentes citadas acima, porém a Escola de Hackers atende apenas cinco escolas municipais de Passo Fundo, seguindo com suas atividades semanalmente em algumas escolas e quinzenalmente em outras. O atual grupo de trabalho é composto por acadêmicos dos cursos de graduação em Psicologia e em Ciência da Computação, bem como do mestrado no Programa de Pós-Graduação em Ensino de Ciências e Matemática, além de professores da área de Psicologia e Computação da Universidade de Passo Fundo. O Programa Escola de

⁷ Atendeu crianças de 5 a 6 anos da Escola Municipal de Educação Infantil (EMEI) Cantinho Feliz e utiliza o programa ScratchJr.

⁸ Aconteceu em dez escolas municipais, atendendo alunos do Ensino Fundamental do 6º ao 9º ano e formou cerca de 150 jovens.

⁹ Destinou-se a pessoas da terceira idade que participam do Departamento de Atenção da Terceira Idade (DATI).

¹⁰ Composta pelos 10 dos melhores programadores da Escola de Hackers e trabalhou com programação em linha de código e, posteriormente, robótica.

Hackers foi agraciado recentemente com o Prêmio Líderes & Vencedores da Assembléia Legislativa do Estado do Rio Grande do Sul.

2.5.2 Organização metodológica do Programa Escola de Hackers

As definições metodológicas são realizadas no início do ano e o acompanhamento e avaliação das mesmas se dá semanalmente pela coordenação do programa com a equipe executora. Tais definições pressupõem a participação ativa do grupo de estudantes de cada escola, articulando-se em torno de dinâmicas não-expositivas que provocam o desenvolvimento da autonomia e do pensamento criativo. Para tanto, utiliza-se a metodologia de aprendizagem baseada em problemas a fim de que, além do aprendizado de programação, os estudantes possam desenvolver competências na área de resolução de problemas, criatividade e trabalho em grupos.

Todas as atividades disponíveis no Caderno Didático¹¹, organizam-se de modo a criar situações problemas onde, além de refletir sobre as melhores formas de resolução, criam oportunidades de aprendizado e exercício de competências na área de programação de computadores. Os módulos, por sua vez, organizam-se em níveis de complexidade crescente e passam, gradativamente, de atividades individuais de conhecimento do *software* utilizado, para trabalhos que pressupõem articulação em grupo e a resolução de problemas.

Os módulos, definidos preliminarmente, são constantemente avaliados pela equipe a fim de detectar possíveis alterações necessárias. Para que se possa efetivar um processo ativo de acompanhamento, os encontros semanais desenvolvem-se em torno da avaliação das atividades nas escolas na semana anterior; da identificação de possíveis alterações na dinâmica proposta; e na preparação coletiva do encontro da semana. Esta etapa conta com a presença da equipe executora e de representantes das IES envolvidas.

Destaca-se que, para a execução da metodologia e dos módulos, é fundamental uma sólida formação dos estagiários, dessa forma, o preparo e estudo é feito semanalmente, nas segundas-feiras. Importante destacar que no momento da seleção, busca-se privilegiar aqueles alunos que já tenham alguma experiência de docência. Uma vez selecionados, propõem-se atividades de formação que passam pelo manuseio dos recursos a serem utilizados nas oficinas da Escola de Hackers como os blocos lógicos e o *software* (Scratch propriamente dito) a ser utilizado nas oficinas.

¹¹ Material utilizado para guiar a aplicação das atividades executadas pelo projeto.

O acompanhamento das atividades ocorre no decorrer do ano de forma semanal, a partir dos relatos dos estagiários nas reuniões de acompanhamento e do preenchimento de formulário específico e constante do sistema de controle de frequência do programa; e mensal, através do preenchimento de um formulário de avaliação por cada um dos estagiários com informações sobre o andamento do programa em cada escola. Além destes dois instrumentos, a equipe organizadora, composta por membros da Secretaria Municipal de Educação e por professores das IES parceiras, realizam visitas periódicas às escolas, em especial àquelas que, em seus relatórios mensais, apresentam situações distintas das demais.

Cabe relatar que se estabeleceu um importante processo de colaboração científica com a Universidade de Passo Fundo que tem vinculado alunos de graduação, mestrado e doutorado para acompanhar as atividades do Programa Escola de Hackers a fim de gerar conhecimento científico oriundo da experiência.

2.5.3 Resultados do projeto em 2017

As análises dos resultados da aplicação do Caderno Didático de 2017, foram realizadas de forma quantitativa com o objetivo de precisar os resultados que foram avaliados partindo dos frutos colhidos dentro de três frentes diferentes: um teste inicial com blocos lógicos, um *software* produzido dentro do Scratch e a evolução dos alunos identificada dentro da plataforma Dr. Scratch.

O **teste com os blocos lógicos** foi utilizado para selecionar os alunos de cinco das dez escolas que participaram do programa em 2017, com o objetivo de verificar se ao fim do ano poderiam se destacar mais e apresentar um percentual menor de desistência de alunos, tornando tal teste pré-requisito para que os indivíduos possam integrar-se ao projeto. Desta forma, a atividade foi iniciada com a identificação dos alunos através de uma etiqueta com seus respectivos nomes, que foi fixada em um lugar visível. Em seguida, os estudantes foram divididos em grupos de cinco componentes e receberam um jogo de blocos lógicos, assim explicamos que cada grupo retiraria uma expressão de cada uma das três caixas com sentenças (Quadro 1), que iam aumentando gradativamente ao término de cada desafio o grau de complexidade. Todos grupos retiraram uma atividade de cada caixa, tendo 5 minutos para executar cada expressão. Manipulando o material disponível (blocos lógicos), a resposta dos problemas deveria ser apresentada com os blocos que correspondessem a sentença para induzir o raciocínio lógico.

Em cada grupo passaram dois avaliadores, um para questionar e ouvir os alunos durante a execução da tarefa e ao término, outro para anotar todas as informações tendo por base os seguintes tópicos: envolveu-se na execução do desafio, destacou-se durante a execução do desafio, demonstrou clareza na apresentação. Cada colocação correta e dentro dos pontos de avaliação valeria um ponto e, ao final, foi somado para obter o resultado.

Quadro 1 - Sentenças utilizadas no teste com blocos lógicos.

CAIXA 1	CAIXA 2
- Blocos circulares e grandes	- Se blocos azuis, então quadrados
- Blocos finos e vermelhos	- Se blocos triangulares, então coloridos
- Blocos azuis ou quadrados	- Se blocos grossos, então circulares
- Blocos amarelos ou grossos	- Se blocos vermelhos, então finos
- Blocos circulares ou grandes	- Se blocos finos, então retângulos
CAIXA 3	
- Se blocos azuis ou vermelhos, então quadrados. Senão, círculos, triângulos e retângulos	
- Se triângulos ou círculos, então colorido. Senão, quadrados amarelos	
- Se blocos grossos, então colorido. Senão, finos e pequenos	
- Se blocos grandes, então amarelos. Senão, finos ou vermelhos	
- Se quadrados ou retângulos, então grossos. Senão, vermelhos	

Fonte: Da autora.

Ao analisar todos os dados referentes ao ano de 2017, detectou-se que os alunos selecionados pelo método de blocos lógicos tiveram um percentual 30% maior de permanência ao final do programa, demonstraram mais interesse, comprometimento, assiduidade e capacidade de desenvolvimento do raciocínio lógico mais apurado, mesmo que tenham apresentado um rendimento diário igualmente satisfatório aos alunos selecionados aleatoriamente pelo corpo diretivo de sua respectiva escola.

No final do ano, os monitores responsáveis por ministrar as aulas do projeto responderam a um questionário e identificaram a satisfação dos resultados apresentados por cada instituição, se as escolas selecionadas com blocos lógicos apresentaram melhores resultados que as demais, e se notaram diferença de comportamento, comprometimento, desempenho, facilidade de compreensão entre os alunos das escolas selecionadas por meio dos blocos lógicos em relação aos demais, justificando suas respostas.

O monitor A destacou que,

tive a experiência em duas escolas, uma com alunos selecionados por blocos lógicos e na outra selecionados pela direção. Na escola com alunos selecionados por blocos lógicos os mesmos demonstraram muito mais interesse e compreensão, absorvendo mais facilmente as atividades propostas, tanto quanto realizando desafios mais rapidamente, outro motivo que também dá a crer que a seleção por blocos lógicos ajuda é o nível de desistências das escolas. [...] Diferença de comportamento: sim, há diferença, os alunos selecionados por blocos se mostram mais comportados, visto que estão mais interessados. [...] Comprometimento: Alunos demonstram um comportamento maior quando selecionados pelos blocos lógicos. [...] Desempenho: Desempenho é superior quando selecionados com blocos lógicos, desenvolvendo as atividades mais rapidamente e com maior facilidade. [...] Facilidade de compreensão: Muita! alunos selecionados com blocos lógicos possuem mais facilidade de compreensão e realizam as atividades mais rapidamente.

O monitor B salientou que, “os alunos são mais interessados e houve um número menor de desistências. Nas primeiras aulas um aluno já havia criado um jogo próprio. Contudo, nas escolas não selecionadas com blocos lógicos os alunos são mais comportados e atentos quanto às explicações”. Ainda criou um comparativo entre a escola selecionada com blocos lógicos e a escola que os alunos foram escolhidos pelo corpo diretivo e docente da instituição. Assim expôs que,

Selecionada com blocos lógicos: Comportamento = Até a metade das aulas eles eram mais comportados e prestavam mais atenção. Na segunda metade em diante começaram a fazer mais bagunça e brincadeiras em aula, principalmente os que foram selecionados para a Escola de Hackers Avançada. Comprometimento = Sempre foram comprometidos, realizaram as tarefas e agora estão criando seus jogos. Alguns ainda têm um pouco de preguiça para fazer algo mais “demorado”, mas a maioria está colaborando. Desempenho = Os que mais se destacavam no início são os que mais estão se destacando agora. Os que menos se destacavam estão melhorando a cada aula, buscando aprender coisas novas. Alguns não possuem um bom desempenho, mas se esforçam para poder realizar as atividades corretamente. Facilidade de compreensão = Boa parte da turma consegue entender e aprender rapidamente. Não selecionada com blocos lógicos: Comportamento = Alunos comportados, atenciosos e dedicados. Comprometimento = Alunos comprometidos, sempre buscaram resolver as atividades, porém houveram várias desistências. Desempenho = Alunos possuem um bom desempenho, tanto ao resolver as atividades quanto ao desenvolver o jogo final. Destaco ainda, que com as desistências, os que restaram ainda buscam “evoluir” a cada aula, aprendendo coisas novas e realizando mais rapidamente as atividades. Facilidade de compreensão = Alguns tem dificuldades, demoram mais para compreender. Outros entendem facilmente e buscam resolver as atividades rapidamente.

Desta maneira, percebeu-se a eficiência do teste com blocos lógicos, pois o índice de permanência dos alunos no projeto foi maior, os quais demonstraram-se mais interessados, comprometidos, assíduos e incentivados. Assim, é prudente que em iniciativas semelhantes ao Programa Escola de Hackers, seja adotada a prática de seleção por blocos lógicos.

Para dar início à análise de tempo de resposta, no intuito de aferir a capacidade de retenção de informação dos participantes do programa e para suprir a necessidade de utilizar *softwares* comerciais para isto, já que muitas vezes tais ferramentas são caras e sua utilização

não é simples, foi criado um *software*¹² usando o Scratch, nomeado por A Prova de Hackers. O *software* criado foi aplicado no início das atividades e tinha previsão de nova aplicação ao final do ano letivo. Tinha como objetivo identificar capacidades cognitivas, em especial o tempo de resposta e a memória de curto prazo dos alunos, para assim ter uma ideia de como o programa estava colaborando para o desenvolvimento dos alunos nestes quesitos. Desta forma, os resultados foram exportados através de um *software* desenvolvido em Java e guardados em uma tabela compartilhada entre os organizadores e monitores do programa.

O *software* A Prova de Hackers consistiu em um questionário interativo, composto por telas de questões (Figura 3), telas de imagens para visualização temporizada do contexto a ser analisado para responder à questão (Figura 4) e telas com opções de resposta (Figura 5). Em seu interior, havia um cronômetro para o tempo de cada questão. Também havia uma lista, que armazenava o nome do aluno, as suas respostas e tempos decorridos para cada questão.

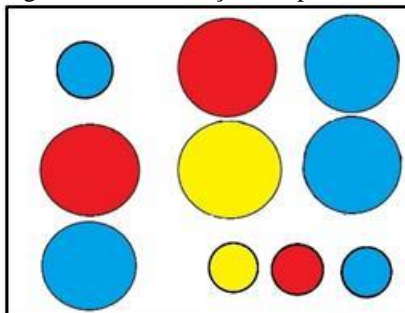
Quando as dez perguntas foram respondidas, a lista com os dados coletados foi exportada do Scratch manualmente e todas as listas foram importadas para uma tabela (Figura 6), modificando o nome dos alunos para assegurar sua identidade.

Figura 3 - Modelo de questões.



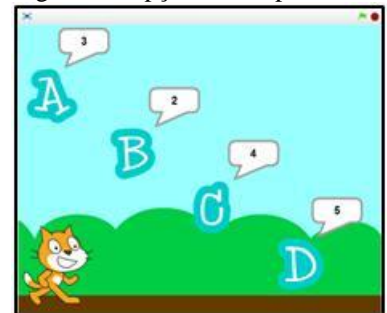
Fonte: <https://scratch.mit.edu/>.

Figura 4 - Visualização temporizada.



Fonte: <https://scratch.mit.edu/>.

Figura 5 - Opções de resposta.



Fonte: <https://scratch.mit.edu/>.

Figura 6 - Lista de respostas das questões.

Nome	SEXO	R	Tempo	R	Tempo	R	Tempo	R	Tempo	R	Tempo	R	Tempo	R	Tempo	R	Tempo	R	Tempo	total:	V:	MÉD TOT:		
Aluno 1	F	F	5,133	V	6,833	F	2,667	V	4,968	V	2,101	V	1,968	F	2,900	F	2,100	V	3,934	V	1,499	34,103	6	3,410
Aluno 2	F	V	3,466	F	34,575	V	3,066	F	1,799	V	2,272	V	1,433	F	3,467	F	7,667	V	1,168	V	1,600	60,513	6	6,051
Aluno 3	F	F	5,469	F	13,834	F	1,300	V	0,933	V	1,167	V	1,433	F	1,000	V	3,200	F	2,700	V	1,100	32,136	5	3,214
Aluno 4	F	F	1,800	F	13,400	F	1,563	F	2,334	V	1,633	V	2,333	F	3,867	F	4,300	V	1,267	V	3,066	35,563	4	3,556
Aluno 5	F	F	6,200	F	9,534	V	2,633	V	1,633	F	1,467	V	2,333	F	2,467	F	3,067	V	2,433	V	1,900	33,667	5	3,367
Aluno 6	F	F	3,534	F	14,233	F	2,667	V	1,400	F	1,433	F	1,633	F	2,567	F	7,568	V	0,700	V	2,500	38,235	3	3,824
Aluno 7	F	V	3,200	V	2,066	F	9,334	V	2,633	V	1,133	V	2,267	F	5,267	F	2,833	V	15,267	V	1,666	45,666	7	4,567
Aluno 8	F	F	1,733	V	5,233	V	2,599	F	4,465	V	1,400	V	1,833	F	2,168	V	3,367	V	2,732	V	2,166	27,696	7	2,770
Aluno 9	F	V	47,267	F	11,900	V	29,233	F	7,967	V	1,533	F	2,234	F	3,633	V	1,900	V	1,300	V	1,300	108,267	6	10,827
Aluno 10	F	F	4,866	V	2,133	F	13,034	V	4,599	V	1,532	V	1,799	F	17,867	V	1,700	F	6,332	V	1,299	55,161	6	5,516
Aluno 11	F	F	23,877	F	11,101	V	2,928	F	19,835	V	2,167	V	2,300	F	6,967	V	2,233	V	2,865	V	2,200	76,473	6	7,647
Aluno 12	F	F	3,466	F	9,867	V	5,632	F	3,799	V	6,300	V	2,500	F	1,199	V	1,934	F	2,432	V	1,068	38,197	5	3,820
Aluno 13	F	F	8,800	F	53,534	F	6,266	V	1,567	V	1,600	V	2,567	F	1,733	V	1,600	V	1,633	V	1,468	80,768	6	8,077
Aluno 14	F	F	5,934	V	9,300	F	5,967	F	2,366	F	1,532	V	1,700	V	2,634	V	5,166	F	5,999	V	1,133	41,731	5	4,173
Aluno 15	F	V	3,033	F	2,400	V	2,200	V	2,200	V	1,734	V	5,900	V	3,166	V	1,900	V	5,767	V	1,733	30,033	9	3,003
Aluno 16	F	F	7,033	F	9,267	V	3,632	V	1,399	V	1,568	V	2,666	F	11,866	V	1,886	V	2,600	V	1,567	43,484	7	4,348
Aluno 17	F	F	11,033	F	14,999	F	2,766	V	10,999	V	2,400	V	2,900	F	2,166	F	3,067	F	3,199	V	3,534	57,063	4	5,706
Aluno 18	F	V	26,233	F	21,234	V	1,833	V	1,767	V	2,466	V	1,567	F	1,167	V	1,369	V	1,333	V	0,967	59,936	8	5,994
Aluno 19	F	F	12,434	F	11,434	F	2,032	V	1,600	F	2,132	V	2,367	V	2,717	F	2,565	V	1,434	V	1,866	40,581	5	4,058
Aluno 20	F	V	2,800	F	9,933	V	2,308	V	1,300	V	1,066	F	1,134	F	7,666	V	0,967	V	1,267	V	0,966	29,407	7	2,941

Fonte: Da autora.

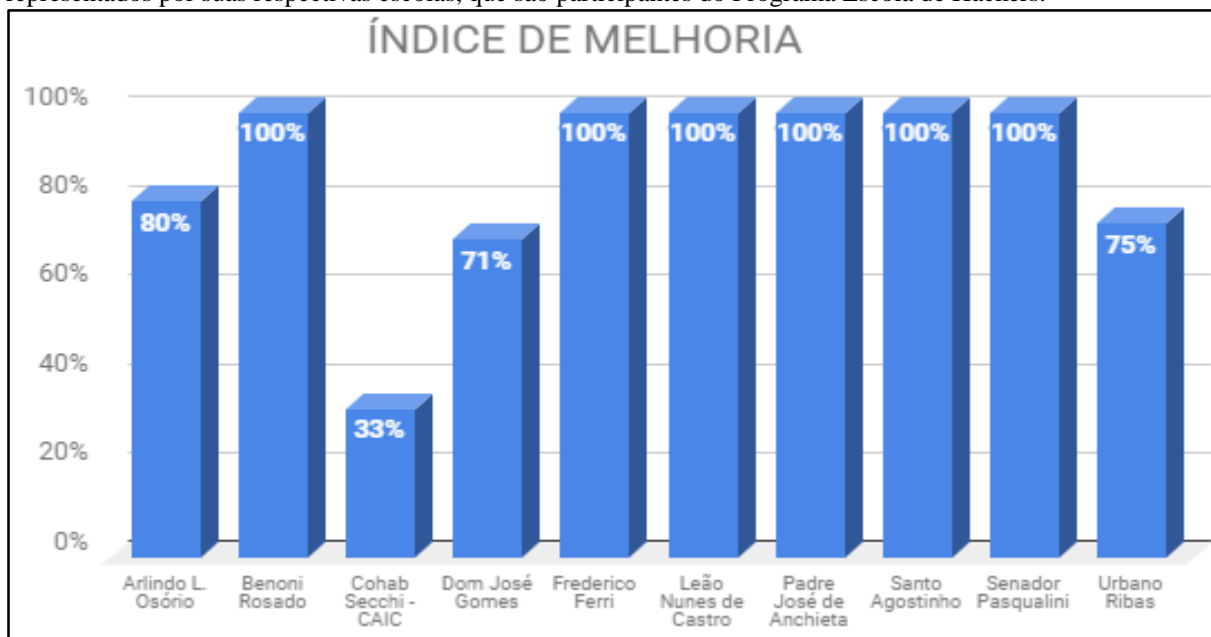
¹² O *software* A Prova de Hackers pode ser acessado no Scratch: <https://scratch.mit.edu/projects/159196859/>.

Assim, foi coletado o nome do aluno, o número da questão, se a resposta foi certa ou errada e o tempo de resposta em segundos e milissegundos. Na tabela de respostas também foi calculada a média individual de tempo, a média geral de tempo e a porcentagem de acertos e erros individuais e gerais. Ao final, foram gerados gráficos dinâmicos sobre os dados, facilitando e agilizando posteriormente o entendimento dos dados coletados.

De acordo com os testes realizados, a média de tempo de todas as questões para as meninas foi de 40,763 segundos com 60,615% de acertos. Já para os meninos a média foi de 33,418 segundos com 64,156% de acertos. Ao final do ano, quando o teste voltou a ser aplicado, foi detectado que 78,16% dos alunos participantes do programa melhoraram seu tempo de resposta, aumentando significativamente seus acertos nas questões propostas. Tendo em vista os resultados quantitativos, pode-se afirmar que o Programa Escola de Hackers tem auxiliado seus participantes a desenvolver suas capacidades cognitivas, apresentando um tempo de resposta mais apurado, além de um maior desenvolvimento da habilidade de memorização.

Com a coleta de dados finalizada, os alunos participantes do programa foram divididos em suas respectivas escolas, contendo números totais sobre a melhoria na capacidade de fixação de conhecimento dos estudantes. Portanto, pode-se concluir que 90% das escolas integrantes melhoraram significativamente a aptidão de retenção de informação de seus alunos, que apresentaram mais acertos em um tempo menor, como mostra a Figura 7.

Figura 7 - Gráfico de apresentação da melhoria na capacidade de retenção de informações dos alunos, representados por suas respectivas escolas, que são participantes do Programa Escola de Hackers.



Fonte: Da autora.

Tratando-se das escolas Arlindo Luis Osório, Benoni Rosado, Dom José Gomes, Frederico Ferri, Padre José de Anchieta e Urbano Ribas, onde ocorreram atividades semanais de 2h, contando com a participação, o apoio e interesse do corpo diretivo e docente de cada instituição para execução do projeto, contribuindo para manter seus alunos interessados, incentivados, participativos e assíduos. Desta forma, detectou-se que dos cinco alunos da Arlindo Luis Osório, quatro melhoraram seu desempenho tanto em tempo como em acertos e um deles manteve a sua performance. Para as escolas Benoni Rosado (seis participantes), Frederico Ferri (dois participantes) e Padre José de Anchieta (cinco participantes), constatou-se que todos os integrantes do programa aperfeiçoaram significativa sua capacidade de reter informações, onde aumentaram seus acertos em um tempo menor. A escola Dom José Gomes, apresentou melhora de tempo e acertos em cinco dos sete alunos participantes, os outros dois mantiveram a quantidade de acertos, apresentando tempo semelhante nos dois momentos. A escola Urbano Ribas tinha quatro educandos associados ao projeto, assim, exibiu avanço em três alunos e um desempenho igual no quesito acertos com um tempo equivalente em um estudante.

Explanando as escolas Cohab Secchi - CAIC, Santo Agostinho e Senador Pasqualini, onde também ocorriam aulas semanais de 2h, porém não contaram com a participação, apoio e interesse ativo do corpo diretivo e docente de cada instituição para realizar o projeto, não ocorrendo comunicação entre escola, monitor e equipe executora do programa, ocasionando muitas desistências por falta de interesse, incentivo e assiduidade. Houveram semanas em que as escolas promoviam outras atividades no horário das oficinas, prejudicando o andamento do conteúdo e aprendizagem dos alunos vinculados ao projeto. Desta forma, observou-se que dos três alunos vinculados a escola Cohab Secchi - CAIC, apenas um deles melhorou seu desempenho em acertos com um tempo menor de execução, e os outros dois mantiveram seus acertos com um tempo parecido. Para as escolas Santo Agostinho (um participante) e Senador Pasqualini (um participante) constatou-se que ambos aperfeiçoaram significativa sua capacidade de reter informações, onde aumentaram seus acertos em um tempo menor.

Explicitando a escola Leão Nunes de Castro, onde ocorriam oficinas quinzenais de 4h, pois tratava-se de uma instituição do interior e com dificuldade de acesso tanto dos alunos como dos monitores. Contou com a participação, o apoio e interesse do corpo diretivo e docente para execução do projeto, contribuindo para manter seus alunos interessados, incentivados, participativos e assíduos. Desta maneira, identificamos que os quatro alunos integrantes do programa melhoraram significativamente sua capacidade de reter informações, apresentando mais acertos com tempo expressivamente menores.

Já a **plataforma Dr.Scratch**¹³, uma ferramenta web gratuita de código aberto, que está sendo desenvolvida por uma equipe de professores, desenvolvedores e pesquisadores da *Universidad Rey Juan Carlos*, na Espanha, analisa automaticamente projetos que foram feitos dentro do Scratch. Esta ferramenta permite que se descubra um bom programador e o ajuda a atualizar habilidades de codificação.

Ao importar um projeto Scratch para o Dr. Scratch, ele fornecerá informações sobre o desenvolvimento do pensamento computacional que o autor obteve, concedendo resultados com valor em relação à capacidade de abstração, pensamento lógico, sincronização, paralelismo, controle de fluxo, interatividade do usuário e representação de dados, podendo atingir a pontuação máxima de 21 pontos, pois cada categoria elencada vale 3 pontos, assim como a Figura 8 abaixo apresenta.

Figura 8 - Elementos medidos pelo software Dr. Scratch.



Fonte: Dr. Scratch (<http://www.drscratch.org>).

Partindo de tais perspectivas, o ano letivo foi organizado em três momentos, visto que três produções individuais, dentro do Scratch, que foram executadas através de desafios lançados no decorrer do programa e foram selecionadas. Para o primeiro momento, optou-se pelo projeto inicial executado, para o segundo momento, um projeto intermediário foi escolhido e para o terceiro momento, o último projeto foi adotado.

Desse modo, as propostas de cada aluno foram executadas na ferramenta Dr.Scratch, a partir da URL¹⁴ dos programas criados individualmente, dentro do Scratch, a fim de quantificar o desenvolvimento de cada aluno participante do programa, buscando analisar se

¹³ Disponível para acesso em: <http://www.drscratch.org/>.

¹⁴ Abreviatura inglesa correspondente a *Uniform Resource Locator*, que foi traduzido para a língua portuguesa como Localizador Uniforme de Recursos. Um URL se refere ao endereço de rede no qual se encontra algum recurso informático.

tal indivíduo estava apresentando evolução ou resultados iguais no decorrer das atividades. Esta análise quantitativa ocorreu dentro das categorias fornecidas pelo *software* utilizado, ou seja, utilizando de critérios como abstração, pensamento lógico, sincronização, paralelismo, controle de fluxo, interatividade do usuário, representação de dados e a pontuação total, assim como mostra o Quadro 2 a seguir, que foi desenvolvida para organizar os dados.

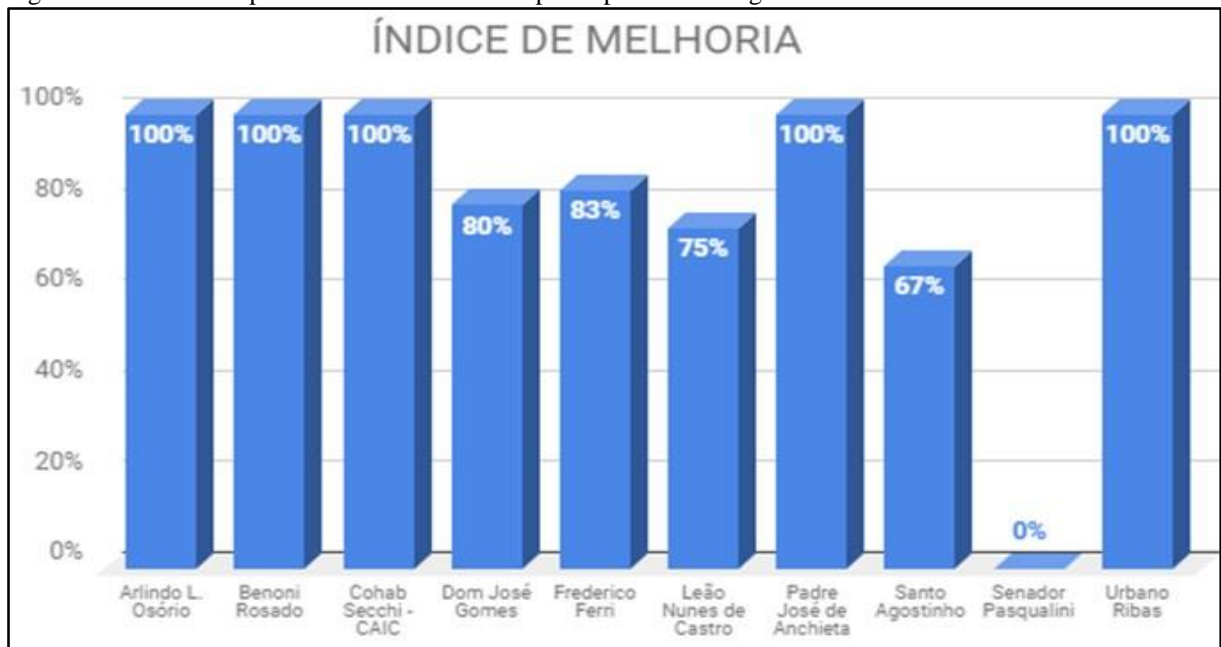
Quadro 2 - Resultados gerados pelos três momentos de cada aluno.

ALUNO 1	L	P	IU	RD	CF	S	A	TOTAL	LINK DO SCRATCH
MOMENTO 1	1	0	2	2	1	2	0	8	https://scratch.mit.edu/projects/167688741/
MOMENTO 2	3	0	2	2	3	2	3	15	https://scratch.mit.edu/projects/170438862/
MOMENTO 3	3	3	2	2	3	3	3	19	https://scratch.mit.edu/projects/171484871/

Fonte: Da autora.

Desta forma, os alunos participantes do programa foram divididos em suas respectivas escolas contendo números totais. Portanto, pode-se afirmar que 80% das escolas integrantes melhoraram seus índices no decorrer do ano, baseando-se na evolução ou involução de cada estudante, conforme os dados fornecidos pelo Dr. Scratch em cada um dos três momentos, como mostra a Figura 9 gráfico a seguir:

Figura 9 - Gráfico de aproveitamento das escolas participantes do Programa Escola de Hackers.



Fonte: Da autora.

Considerando as características¹⁵ já citadas das escolas participantes, percebe-se que na maioria delas os alunos obtiveram resultados superiores a 65%, ou seja, aprimoraram os

¹⁵ Tais características foram explanadas nas páginas 36 e 37 desta dissertação.

seus índices no decorrer do ano e tratando-se da escola Senador Pasqualini, afirma-se que a performance abaixo das demais instituições está diretamente ligada a falta de periodicidade das atividades do projeto, pois a mesma priorizava tarefas extraclasse nos horários em que deveriam ocorrer as aulas.

Com os dados levantados a partir das três fontes citadas acima, é possível afirmar que o programa vem alcançando seus principais objetivos, pois desde a sua concepção busca trazer para seus participantes o desenvolvimento de habilidades que vão além da programação de computadores, ou seja, os alunos devem conseguir desenvolver seu raciocínio lógico, capacidade de memorização e interpretação, ao ponto de obter melhores resultados ao final do ano.

Neste ano de 2018 pretende-se continuar com as quatro frentes do programa: Berçário de Hackers, Escola de Hackers, Escola de Hackers Avançada e Academia White Hat. Atualmente, têm-se 5 escolas participantes e a expectativa de trabalhar com 110 alunos, que serão selecionados por meio do teste com blocos lógicos, para que os componentes cheguem com uma propensão maior de desenvolver o raciocínio lógico e, posteriormente, o pensamento computacional.

No primeiro dia de trabalho, todos os integrantes escolhidos irão responder o *software* genérico e, ao final do ano, o processo será repetido, de modo a gerar dados de comparação, com o objetivo de aferir o desenvolvimento da capacidade cognitiva dos alunos, em especial o tempo de resposta e a competência de retenção de informação a curto prazo. Por fim, serão selecionados três projetos, ou seja, o inicial, o intermediário e o final de cada aluno para passar pela análise da plataforma Dr. Scratch, gerando dados que serão analisados quanto aos objetivos do programa, ou seja, quanto ao desenvolvimento do raciocínio lógico, capacidade de memorização e interpretação dos alunos.

Os novos resultados são esperados com expectativa de melhora expressiva, pois o Caderno Didático, já aplicado em 2017, foi adaptado a partir da Taxonomia de Bloom, voltada para a formação educacional. Além disso, tem como objetivo a aprendizagem visando o desenvolvimento cognitivo e não somente por processos de transferência de conteúdo.

Tal proposta será reaplicada em todas as escolas participantes do programa. Para tanto, os alunos de uma escola foram acompanhados presencialmente a partir de um processo de observação não-participante das oficinas semanais. Ao final do processo o Caderno Didático será hospedado em um blog (Anexo A), que servirá como material de apoio aos professores na estruturação de suas aulas que procuram abordar a programação de computadores.

3 O ENSINO DE MATEMÁTICA PARA O DESENVOLVIMENTO DO RACIOCÍNIO LÓGICO

Neste capítulo tem-se por objetivo ilustrar a forma como o ensino de Matemática está pautado dentro dos Parâmetros Curriculares Nacionais (PCN), seus objetivos, as habilidades a serem desenvolvidas e os ciclos que devem estar contidos no processo de ensino e aprendizagem. Destaca-se que a finalidade do capítulo é oferecer ao leitor clareza sobre o que é trabalhado nas escolas, dentro da disciplina de Matemática, destacando um ponto de grande importância tratando-se de ensino e programação, o raciocínio lógico. Dentro deste contexto propõe-se a construção cognitiva pautada pela taxonomia dos objetivos educacionais, ampliando a teoria visando a utilização da tecnologia, para o desenvolvimento do raciocínio lógico.

3.1 Ensino de Matemática segundo os PCN

Segundo as pautas dos PCN do Ensino Fundamental, a Matemática fornece instrumentos eficazes para compreender e atuar no mundo, ela é uma ferramenta essencial na solução de vários tipos de problemas. Nela são desenvolvidas estruturas abstratas baseadas em modelos concretos; além de método, a Matemática é um meio de comunicação, uma linguagem formal e precisa que requer uma prática constante de forma clara e universal. O conhecimento matemático faz parte do patrimônio cultural da humanidade porque possui características e procedimentos próprios que também tem evoluído no contexto de outras ciências.

Para João Thomaz Pereira,

formar cidadãos preparados para um mundo contemporâneo é um grande desafio para quem dimensiona e promove a educação. Em plena Era do Conhecimento, na qual inclusão digital e Sociedade da Informação são termos cada vez mais frequentes, o ensino não poderia se esquivar dos avanços tecnológicos que se impõem ao nosso cotidiano (2005, p. 13).

A Matemática é componente importante na construção da cidadania, nos conhecimentos científicos e recursos tecnológicos, e o seu ensino deve ser meta prioritária do trabalho docente, procurando desenvolver nos alunos competências para compreender e transformar a realidade. No ensino da Matemática destacam-se aspectos básicos como relacionar observações do mundo real com representações (esquemas, tabelas, figuras) que

devem relacionar-se com princípios e conceitos matemáticos, através da “fala” e da “escrita”. A aprendizagem em Matemática está ligada à compreensão, isto é, à assimilação do significado resultante das conexões entre todas as disciplinas com o cotidiano nos seus diferentes temas.

Diante dessa situação, alguns questionamentos são postos: por que ensinar Matemática? Para quê? A Matemática é uma das mais importantes ferramentas da sociedade moderna, ela contribui para a formação do futuro cidadão que se engajará no mundo do trabalho, das relações sociais, culturais e políticas. Para exercer plenamente a cidadania, é preciso saber contar, comparar, medir, calcular, resolver problemas, construir estratégias, comprovar e justificar resultados, argumentar logicamente, conhecer formas geométricas, organizar, analisar e interpretar criticamente as informações, além de conhecer formas diferenciadas de abordar problemas, assim como Alves (2003) apresenta, “o que diferencia a espécie humana das demais espécies do reino animal é a capacidade de pensar, raciocinar, formular hipóteses, representar mentalmente situações, operar sobre uma situação inicial visando uma situação desejada, enfim, solucionar problemas”.

A Matemática, vista como uma maneira de pensar, como um processo em permanente evolução (não sendo algo pronto e acabado que apenas deve ser estudado), permite, dinamicamente, por parte do aluno, a construção e a apropriação do conhecimento. Ensinar Matemática é importante porque ela se faz presente em todas as coisas, com maior ou menor complexidade. Perceber isso é compreender o mundo e poder atuar nele como cidadão. Em casa, na rua, nas várias profissões, na cidade, no campo, nas várias culturas, o ser humano necessita da Matemática.

Em uma sociedade voltada ao conhecimento e à comunicação, como a do terceiro milênio, é preciso que as crianças aprendam a comunicar ideias, executar procedimentos e desenvolver atitudes Matemáticas, falando, dramatizando, escrevendo, desenhando, representando, construindo tabelas, diagramas e gráficos, fazendo pequenas estimativas, conjecturas e inferências lógicas, etc., tudo isso trabalhando individualmente, em duplas ou pequenas equipes, expressando ideias e respeitando o pensamento dos colegas. Novas competências demandam novos conhecimentos; o mundo do trabalho requer pessoas preparadas para utilizar diferentes tecnologias e linguagens (que vão além da comunicação oral e escrita), instalando novos ritmos de produção, de assimilação rápida de informações, resolvendo e propondo problemas em equipe.

O ensino da Matemática desenvolve no aluno a compreensão dos fenômenos que ocorrem no ambiente, como: poluição, desmatamento, limites para uso dos recursos naturais e

desperdício. Além disso, transforma ferramentas essenciais em conceitos (medidas, áreas, volumes, proporcionalidades, etc.) e procedimentos matemáticos (formulação de hipóteses, realização de cálculos, coleta, organização e interpretação de dados estatísticos, prática de argumentação, etc.). O acompanhamento do próprio desenvolvimento físico (altura, peso, musculatura) e os estudos dos elementos que compõem a dieta básica são alguns exemplos de trabalho que podem servir de contexto para se ensinar Matemática.

A Matemática comporta um amplo campo de relações, regularidades e coerências que despertam a curiosidade e instigam a capacidade de generalizar, projetar, prever e abstrair, favorecendo a estruturação do pensamento e o desenvolvimento do raciocínio lógico. Faz parte da vida de todas as pessoas nas experiências mais simples como contar, comparar e operar sobre quantidades. Nos cálculos relativos a salários, pagamentos e consumo, na organização de atividades como agricultura e pesca, a Matemática se apresenta como um conhecimento de muita aplicabilidade. Também é um instrumental importante para diferentes áreas do conhecimento, por ser utilizada em estudos tanto ligados às ciências da natureza como às ciências sociais e por estar presente na composição musical, na coreografia, na arte e nos esportes.

Essa potencialidade do conhecimento matemático deve ser explorada, da forma mais ampla possível, no Ensino Fundamental. Para tanto, é importante que a Matemática desempenhe, equilibrada e indissociavelmente, seu papel na formação de capacidades intelectuais, na estruturação do pensamento, na agilização do raciocínio dedutivo do aluno, na sua aplicação a problemas, situações da vida cotidiana e atividades do mundo do trabalho e no apoio à construção de conhecimentos em outras áreas curriculares.

Segundo os PCN, disponíveis no portal do Ministério da Educação (MEC)¹⁶, o ensino de Matemática de 6º à 9º ano do Ensino Fundamental tem por objetivo, como competências específicas em geral, levar o aluno a:

- adotar uma atitude positiva em relação à Matemática, ou seja, desenvolver sua capacidade de “fazer Matemática” construindo conceitos e procedimentos, formulando e resolvendo problemas por si mesmo, para aumentar sua autoestima e perseverança na busca de solução para um problema;
- pensar logicamente, relacionando ideias, descobrindo regularidade e padrões, estimulando sua curiosidade, seu espírito de investigação e sua criatividade na solução de problemas;

¹⁶ Disponível para acesso em: <http://portal.mec.gov.br/>.

- formular e resolver situações-problema. Para isso, o aluno deverá ser capaz de elaborar planos e estratégias para a solução do problema, desenvolvendo várias formas de raciocínio (estimativa, analogia, indução, busca de padrão ou regularidade, pequenas inferências lógicas, etc.) executando esses planos e essas estratégias com procedimentos adequados;
- comunicar-se de modo matemático, argumentando, escrevendo e representando de várias maneiras (com números, tabelas, gráficos, diagramas, etc) as ideias Matemáticas;
- interagir com os colegas cooperativamente, em dupla ou em equipe, auxiliando-os e aprendendo com eles, apresentando suas ideias e respeitando aquelas que são diferentes, formando assim, um ambiente propício à aprendizagem;
- desenvolver competências para aprender a identificar e buscar os conhecimentos necessários para resolver uma situação-problema.

Nestes ciclos, os Parâmetros Curriculares Nacionais para a área de Matemática no Ensino Fundamental estão pautados por princípios decorrentes de estudos, pesquisas, práticas e debates desenvolvidos nos últimos anos, cujo objetivo principal é o de adequar o trabalho escolar a uma nova realidade, marcada pela crescente presença dessa área do conhecimento em diversos campos da atividade humana. São eles:

- a Matemática é importante na medida em que a sociedade necessita e se utiliza, cada vez mais, de conhecimentos científicos e recursos tecnológicos, que por sua vez são essenciais para a inserção das pessoas como cidadãos no mundo do trabalho, da cultura e das relações sociais;
- a atividade Matemática escolar não é olhar para coisas prontas e definitivas, mas a construção e a apropriação de um conhecimento pelo aluno, que se servirá dele para compreender e transformar sua realidade;
- o ensino de Matemática deve garantir o desenvolvimento de capacidades como: observação, estabelecimento de relações, comunicação (diferentes linguagens), argumentação e validação de processos e o estímulo às formas de raciocínio como intuição, indução, dedução, analogia, estimativa;
- o ensino-aprendizagem de Matemática tem como ponto de partida a resolução de problemas;
- no ensino da Matemática, destacam-se dois aspectos básicos: um consiste em relacionar observações do mundo real com representações (esquemas, tabelas,

figuras, escritas numéricas); outro consiste em relacionar essas representações com princípios e conceitos matemáticos. Nesse processo, a comunicação tem grande importância e deve ser estimulada, levando o aluno a falar e a escrever sobre Matemática, a trabalhar com representações gráficas, desenhos, construções e a aprender como organizar e tratar dados;

- o conhecimento matemático é historicamente construído e, portanto, está em permanente evolução. Assim, o ensino de Matemática precisa incorporar essa perspectiva, possibilitando ao aluno reconhecer as contribuições que ela oferece para compreender as informações e posicionar-se criticamente diante delas;
- recursos didáticos como livros, vídeos, televisão, rádio, calculadoras, computadores, jogos e outros materiais têm um papel importante no processo de ensino e aprendizagem. Contudo, eles precisam estar integrados a situações que levem ao exercício da análise e da reflexão.

Para o desenvolvimento das habilidades previstas para o Ensino Fundamental – Anos Finais, é imprescindível levar em conta as experiências e os conhecimentos matemáticos já vivenciados pelos alunos, criando situações nas quais possam fazer observações sistemáticas de aspectos quantitativos e qualitativos da realidade, estabelecendo inter-relações entre eles e desenvolvendo ideias mais complexas. Essas situações precisam articular múltiplos aspectos dos diferentes conteúdos, visando ao desenvolvimento das ideias fundamentais da Matemática como equivalência, ordem, proporcionalidade, variação e interdependência.

Da mesma forma que na fase anterior, a aprendizagem em Matemática no Ensino Fundamental – Anos Finais também está intrinsecamente relacionada à apreensão de significados dos objetos matemáticos. Esses significados resultam das conexões que os alunos estabelecem entre os objetos e seu cotidiano, entre eles e os diferentes temas matemáticos e, por fim, entre eles e os demais componentes curriculares. Nessa fase, destaca-se a importância da comunicação em linguagem Matemática com o uso da linguagem simbólica, da representação e da argumentação.

Para Justino,

no universo da educação, a utilização de recursos didáticos e da tecnologia inovadora, somados a prática pedagógica adequada, busca despertar o interesse para o aprendizado, pois oferecem um conjunto de recursos importantes e ferramentas de comunicação e informações, tornando-se, assim, um componente essencial de pesquisa e um potente instrumento de ensino-aprendizagem (2011, p. 73).

Diferentes recursos didáticos e materiais, como malhas quadriculadas, ábacos, jogos, calculadoras, planilhas eletrônicas e *softwares*, precisam estar integrados a situações que propiciem a reflexão, contribuindo para a sistematização e a formalização dos conceitos matemáticos. Cumpre também considerar que, para a aprendizagem de certo conceito ou procedimento, é fundamental haver um contexto significativo para os alunos, não necessariamente do cotidiano, mas também de outras áreas do conhecimento. No entanto, é indispensável que eles desenvolvam a capacidade de abstrair o contexto, apreendendo relações e significados para aplicá-los em outros contextos. Assim, pretende-se que os alunos formulem novos obstáculos, baseando-se na reflexão e no questionamento sobre o que ocorreria se alguma condição fosse modificada ou se algum dado fosse acrescentado ou retirado do problema proposto. Além disso, na fase final do Ensino Fundamental é importante iniciar com os alunos, gradativamente, a compreensão, análise e avaliação da argumentação Matemática. Isso envolve a leitura de textos matemáticos e o desenvolvimento do senso crítico em relação à argumentação neles utilizada.

Desta forma, ao potencializar o ensino de programação nas escolas é incentivar professores e estudantes a criar aplicativos para a resolução de problemas sociais que envolvam a comunidade onde vivem. São ideias simples e inovadoras de como abordar o uso da tecnologia no processo de ensino-aprendizagem de uma forma mais dinâmica e interativa, que poderiam contemplar a atual proposta de Base Nacional Comum Curricular, especialmente quando trata do desenvolvimento do pensamento computacional, tendo em vista a capacidade dos alunos de interpretar situações problemas, investigar, decompor em partes mais simples e ordená-las conforme a construção mental, para obter seus resultados através da programação de computadores, assim estarão aptos a explicá-los de maneira clara e argumentos concretos.

Com as mudanças tecnológicas e informacionais que a sociedade sofreu nos últimos anos, é cada vez mais necessário que as instituições de ensino incorporem à sua grade consolidada novas matérias que atendam as demandas do século XXI, sendo o ensino de programação nas escolas uma das mais importantes.

Todas as atividades realizadas com os alunos deverão servir de suporte para educar cidadãos mais capazes de usar seu raciocínio lógico para realizar trabalhos coerentes, com senso crítico e analítico, pois entende-se que é na escola que o indivíduo é preparado para atuar no mundo e, é também nela, que o sujeito constrói a interação, onde práticas sociais acontecem.

3.2 Raciocínio lógico

O cenário atual no qual os processos de ensino e aprendizagem estão inseridos é dinâmico, marcado pelo encurtamento de fronteiras em função do desenvolvimento de novas tecnologias, onde as informações podem ser adquiridas e processadas quase que instantaneamente. Por isso, faz-se necessária a compreensão de tais informações pelo sujeito, de tal modo que este possa selecioná-las e entendê-las, diagnosticando seu contexto social, político e cultural. Por essa razão, este sujeito necessitará desenvolver o domínio de ferramentas que analisem as informações de forma crítica e construtiva. São os processos mentais que irão modular a informação recebida, a fim de dar soluções e possíveis esclarecimentos àquilo apresentado, pois o ser humano precisa pensar, buscar formas cognitivas superiores para construir os significados. Dessa forma, pode-se afirmar que se trata de um raciocinar que busca sistematizar o pensamento em relação à informação, tal processo é conhecido como Raciocínio Lógico, ou segundo Piaget (1978), o “conhecimento lógico matemático”.

Dante afirma que, “é preciso desenvolver no aluno a habilidade de elaborar um raciocínio lógico e fazer uso inteligente e eficaz dos recursos disponíveis, para que ele possa propor boas soluções às questões que surgem em seu dia-a-dia, na escola ou fora dela” (2000, p. 11-12).

Pode-se dizer que o raciocínio lógico surgiu concomitantemente com o nascimento da Matemática. O pensar lógico e a Matemática sempre estiveram intimamente ligados, mesmo que tenham sido formuladas de modo separado e tenham se constituído como ciência de forma independente. Afirma-se, então, que a Matemática estabelece laços fortes com a lógica a partir do princípio da argumentação, com base em critérios formulados e analisados pelo ser humano, deixando de ser subjetivo, mesmo que use a dedução como premissa para algumas formulações, passando a ter caráter expositivo de forma organizada, seja por símbolos ou pela técnica, como os algoritmos.

Assim, a Matemática e a lógica não são ciência das verdades vistas como absolutas e únicas, mas sim ferramentas de análise, instrumentos que auxiliam na forma de pensar e estruturar o raciocínio, levando à argumentação. Dessa forma, argumentar não é atividade exclusiva da Matemática. Na realidade, a boa demonstração, em âmbitos gerais, depende basicamente da apropriação adequada da língua materna e de estímulos adequados, ou seja, argumentar é questão de prática. De fato, embora não seja possível defender que o estudo da Matemática tenha alguma primazia sobre o desenvolvimento do raciocínio lógico, também

não se pode deixar de constatar que a Matemática é realmente um terreno muito promissor para isso. E, por outro lado, sem o uso do raciocínio lógico, é difícil chegar ao cerne das questões Matemáticas.

O raciocínio lógico está ligado a conceitos capazes de organizar e clarear as situações cotidianas, preparando os jovens para circunstâncias mais complexas. De acordo com o Construtivismo, a Matemática ensinada através da imposição de fórmulas, exercícios repetitivos e conceitos limitados, impossibilita o aprendizado, gerando alunos passivos, desinteressados e com falta de criatividade. Assim como Faria afirma,

a experiência física está relacionada à experiência lógico-matemática; uma é condição para o aparecimento da outra. Para uma pessoa realizar uma experiência lógico-matemática, como a de reunir ou classificar objetos, precisa ter descoberto, anteriormente, as características semelhantes e diferentes dos mesmos, ou seja, ter realizado uma operação física. O processo construtivo da experiência lógico-matemática inicia-se no plano da ação concreta, mas, aos poucos, há uma substituição da ação afetiva pela simbolização, como consequência do desenvolvimento e da tomada de consciência (1998, p. 3-14).

A utilização do raciocínio lógico na formação educacional de jovens gera pessoas críticas com senso argumentativo, e é com essa característica que os alunos desenvolvem a capacidade de criar, interpretar, responder e explicar situações problemas envolvendo Matemática. A utilização desse recurso metodológico influi em resultados positivos, contribuindo em três aspectos básicos: ler, escrever e resolver problemas. Esses que, após a sequência de estudos lógicos, passam a representar novas sistematizações: aprender a ler bem, aprender a escrever bem e aprender a resolver problemas matemáticos bem, de acordo com vários educadores que trabalham com a metodologia das atividades lógicas. Assim como afirma Scolari et al, “da mesma forma que na leitura ou escrita, o raciocínio lógico na resolução de problemas matemáticos é um fator de extrema importância. É fundamental que os alunos compreendam e raciocinem sobre o que está sendo proposto e não somente decorem e apliquem fórmulas” (2007, p. 3).

Na atualidade, os trabalhos com o raciocínio lógico estão ligados a análise de problemas, onde são elaborados exercícios de contexto ou que tenham rigor formal dos símbolos para se aplicar os princípios da lógica Matemática, usando todas as simbologias, métodos, implicações, operações e tabelas que esta possui enquanto disciplina. A área da computação também se apossou de tais ciências, mostrando que a aplicação da lógica, agora denominada de lógica computacional, é inerente para a sua formulação e execução, tendo como resultados uma evolução tecnológica de grande influência para a humanidade.

Considerando o processo de aprendizagem em ambiente computacional, vale ressaltar que o uso da programação de computadores pode promover pequenos e incessantes saltos qualitativos no desenvolvimento do pensamento humano, pois inúmeras situações exigem o emprego de raciocínio abstrato bem como de inferências lógicas de maneiras mais rápidas e eficazes do que ocorrem em outros ambientes de aprendizagem. Sobre o desenvolvimento de estratégias na procura de soluções para situações ao aprender programação, Resnick assim se posiciona,

no processo de aprender a programação, as pessoas aprendem muitas outras coisas. Eles não estão apenas aprendendo uma programação, eles estão programando para aprender. Além de aprender noções Matemáticas e Computacionais (como variáveis e condições), eles também estão aprendendo estratégias para a resolução de problemas, elaboração de projetos e comunicar idéias. Essas habilidades são úteis não apenas para os cientistas da computação, mas para todos, independentemente da idade, interesses ou ocupação (2013, p. s/n).

Neste contexto, percebe-se que trabalhar através do lúdico, acessando recursos didáticos é a forma mais interessante e dinâmica para o desenvolvimento do raciocínio lógico, memorização, capacidade de estratégia e resolução de problemas, além de influenciar diretamente na formação de cidadãos expressivos e capazes de argumentar diferentes situações.

Nesse sentido, o lúdico pode contribuir de forma significativa para o desenvolvimento do ser humano, seja ele de qualquer idade, auxiliando não só na aprendizagem, mas também no desenvolvimento social, pessoal e cultural, facilitando o processo de socialização, comunicação, expressão e construção do pensamento. Vale ressaltar, porém, que o lúdico não é a única alternativa para a melhoria no intercâmbio de ensino aprendizagem, mas é uma ponte que auxilia no aperfeiçoamento dos resultados por parte dos educadores interessados em promover mudanças. Portanto, as atividades lúdicas permitem experimentar, sentir, confrontar real e imaginário, criar e recriar mundos e situações. Através delas, as pessoas se libertam da automatização e vão muito além deste mundo, trocam experiências, vivem momentos de alegria e liberdade, enfim, aprendem com as situações.

3.3 Construção cognitiva pautada pela taxonomia de Bloom

Esta seção tem o propósito de apresentar a taxonomia dos objetivos educacionais, desenvolvida por Benjamin Samuel Bloom e seus colaboradores, além de expor a possibilidade de desenvolver a teoria, utilizando-se da tecnologia, para a ampliação do

raciocínio lógico. Desta maneira, a sequência abordada fará uso do *software* Scratch para a resolução de problemas matemáticos, que estão contidos no Caderno Didático utilizado pelo Projeto Escola de Hackers.

3.3.1 *Taxonomia dos objetivos educacionais*

Na década de 40 e 50 os sistemas de ensino caminhavam em direção a um padrão que transmitia e recuperava informações, ou seja, a escola era um espaço onde só o professor falava, era “dono” de todo conhecimento abordado e recuperava informações somente através de provas individuais sobre o tema estudado. Prevalecendo assim, a formação tecnicista, de características instrucionais, onde o ensino centralizava na formação do indivíduo, e não no seu desenvolvimento intelectual e pessoal.

Diante deste fato desconfortável para Benjamin Samuel Bloom (1913-1999) em 1948, junto a um grupo de psicólogos interessados em testes de aproveitamento escolar, detectaram a ausência de um esquema de referência para o ensino. Assim, criaram um quadro teórico, denominado taxonomia, visando a facilidade de comunicação entre professores e alunos com o objetivo de desenvolver a aprendizagem, aprimorando os domínios cognitivos, sem centralizar na transferência de conteúdos.

A taxonomia de objetivos educacionais, ou ainda, taxonomia de Bloom, como ficou conhecida após a sua publicação em 1956, é considerada como uma organização para classificar objetivos educacionais e tem sido ampliada há muitos anos por Bloom e outros colaboradores.

De acordo com Moreira (2009), Benjamin S. Bloom graduou-se na Universidade Estadual da Pensilvânia, em 1935. A seguir, em 1940, juntou-se ao Departamento de Educação da Universidade de Chicago e concluiu um doutorado em Educação em 1942. Ao longo do curso de doutorado conheceu seu mentor Ralph Tyler, com quem começou a refinar suas ideias para a ampliação da taxonomia de especificações para permitir a formação educacional e os objetivos de aprendizagem a serem planejados e medidos corretamente, ou seja, melhorar a eficácia do desenvolvimento de “domínio” em vez de simplesmente transferir para recordar fatos irracionais.

Para Mager (1961 apud BLOOM, 1964), “Objetivo é uma intenção comunicada por uma descrição minuciosa que apresenta uma mudança proposta no aprendiz - uma especificação sobre a situação do aprendiz quando ele tiver concluído com sucesso a experiência de aprendizagem”. Desta maneira, conclui-se que os objetivos educacionais são

definições de um padrão de comportamento e também de desempenho, que se espera que o aluno seja necessariamente capaz de atingir e ou manifestar.

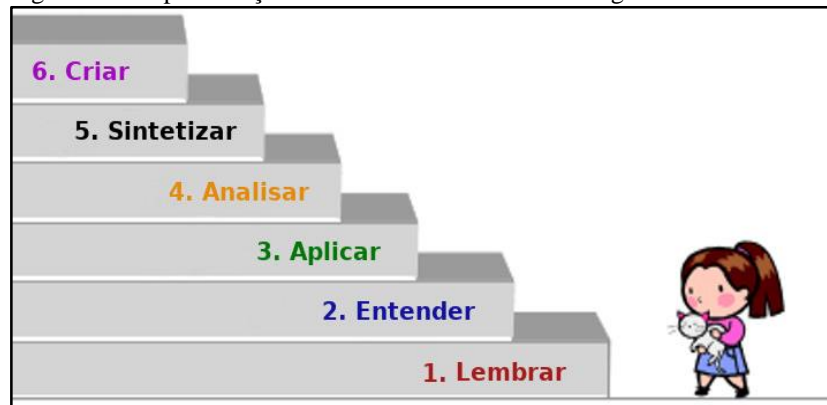
No início, Benjamin S. Bloom imaginava que a educação devia concentrar-se no “domínio” dos indivíduos e do estímulo de modelos superiores de pensamento, ao invés de uma abordagem utilitarista, pautada na transferência de informações. Desta forma, Bloom demonstrou que a maioria dos sistemas de ensino tende a focar na transferência e recuperação de informação, ao contrário do desenvolvimento do verdadeiro significado pessoal, e este continua sendo uma das principais dificuldades para educadores e formadores em tempos modernos.

A Taxonomia de Bloom, desde 1956 concedeu uma base para as ideias que foram utilizadas e desenvolvidas em todo o mundo por acadêmicos, educadores, professores e formadores, para produzir materiais pedagógicos de avaliação. Geralmente estes conceitos que formam o conjunto da Taxonomia de Bloom continuam a ser úteis e muito relevantes para o planejamento e desenho de ensino. Servindo como modelo para a avaliação da aprendizagem dos indivíduos. A categorização proposta por Bloom dividiu as probabilidades de aprendizagem em três grandes domínios: cognitivo, afetivo e psicomotor.

- O **cognitivo**, abrangendo a aprendizagem intelectual, ou seja, tem por objetivo enfatizar, relembrar ou reproduzir algo que foi aprendido, ou que envolve a resolução de alguma atividade intelectual para a qual o indivíduo tem que determinar o problema essencial, então reorganizar o material ou combinar ideias, métodos ou procedimentos previamente aprendidos;
- O **afetivo**, abrangendo os aspectos de sensibilização e graduação de valores, ou seja, tem por objetivo enfatizar o sentimento, emoção ou grau de aceitação ou rejeição. Tais objetivos são expressos como interesses, atitudes ou valores;
- O **psicomotor**, abrangendo as habilidades de execução de tarefas que envolvem o aparelho motor, ou seja, tem por objetivo enfatizar alguma habilidade muscular ou motora.

Desta maneira, trataremos apenas da Taxonomia de Bloom do domínio Cognitivo, que é composta por seis níveis de complexidade crescente, o que significa que para adquirir uma nova habilidade que pertence ao próximo nível, um indivíduo precisa ter compreendido e adquirido a habilidade do nível anterior, como mostra a Figura 10 a seguir.

Figura 10 - Representação dos seis níveis do domínio cognitivo.



Fonte: <https://goo.gl/PsFwj>, adaptado pela autora.

A teoria sustenta-se no fato de que um indivíduo somente pode compreender e aplicar um conceito específico depois de conhecê-lo. Nesta perspectiva, a taxonomia não é simplesmente um padrão de classificação, mas, sim, uma possibilidade de organização hierárquica dos processos cognitivos seguindo os seis níveis de complexidade e objetivos programados.

Segundo Faria,

Deve-se observar que Bloom não pretendeu classificar metodologias de ensino, modos de relacionamento de professores com alunos ou diferentes tipos de materiais de ensino empregado. Não foi interesse de sua equipe classificar um material específico ou conteúdo, mas sim classificar “o comportamento esperado” – a maneira na qual os alunos devem agir, pensar ou sentir como resultado de sua participação em qualquer unidade de ensino (2010, p. 30).

Desta forma, Bloom et al. (1972) e seus colaboradores estavam empenhados em desenvolver uma taxonomia específica para avaliar comportamentos dos alunos diante de diversas situações de aprendizagem. Neste contexto os seus níveis do domínio cognitivo da proposta de Bloom significam:

- **Lembrar:** Refere-se à habilidade do aluno em recordar, definir, reconhecer ou identificar informação específica, a partir de situações de aprendizagem anteriores, ou seja, processos que requerem que o estudante reproduza com exatidão uma informação que lhe tenha sido dada, seja ela uma data, um relato, um procedimento, uma fórmula ou uma teoria;
- **Entender:** Refere-se à habilidade de o aluno em demonstrar compreensão pela informação, sendo capaz de reproduzir a mesma por ideias e palavras próprias, ou seja, requer elaboração (modificação) de um dado ou informação original. O estudante deverá ser capaz de usar esse conhecimento primário e ampliá-lo, reduzi-

lo, representá-lo de outra forma ou prever consequências resultantes da informação original;

- **Aplicar:** Refere-se à habilidade do aluno em recolher e aplicar informação em situações ou problemas concretos, ou seja, reúne processos nos quais o estudante transporta uma informação genérica para uma situação nova e específica;
- **Analisar:** Refere-se à habilidade do aluno em estruturar informação, separando as partes das matérias de aprendizagem e estabelecer relações, explicando-as, entre as partes constituintes, ou seja, caracterizam-se por separar uma informação em elementos componentes e estabelecer relações entre eles;
- **Sintetizar:** Refere-se à habilidade do aluno em recolher e relacionar informação de várias fontes, formando um produto novo, ou seja, representa os processos nos quais o estudante reúne elementos de informação para compor algo novo que terá, necessariamente, traços individuais distintivos;
- **Criar:** Refere-se à habilidade do aluno em fazer julgamentos sobre o valor de algo (produtos, ideias, etc.) considerando critérios conhecidos, ou seja, representa os processos cognitivos mais complexos. Consiste em confrontar um dado, uma informação, uma teoria, um produto etc... com um critério ou conjunto de critérios, que podem ser internos ao próprio objeto de avaliação, ou externos a ele.

A demanda dos níveis cinco (sintetizar) e seis (criar) da Taxonomia de Bloom é fundamental para quem deseja ampliar as capacidades intelectuais, que direcionam a uma resolução criativa de problemas diversos e complexos, que muitas vezes não se apresentam de forma familiar ao indivíduo, que deve ser capaz de reconhecê-los e, obter as soluções mais apropriadas à situação.



É significativo que os objetivos educacionais retratem o nível de conhecimento desejado para os alunos. Assim, o planejamento deve ser elaborado e executado conforme a realidade encontrada, de forma a alcançar tais objetivos. A taxonomia de Bloom tem sido um modelo sistemático seguro, para definir em qual nível se encaixam as metas desejadas para construir bons resultados nas atividades educacionais.



3.3.2 Utilização do Scratch no desenvolvimento de problemas matemáticos


Ser capaz de escrever programas de computador tem implicações diretas sobre o desenvolvimento do raciocínio lógico, uma vez que, programando, os educandos aprendem

estratégias importantes para a resolução de problemas. Nesse sentido, a seguir no Quadro 3 serão apresentadas algumas atividades, que estão contidas no Caderno Didático utilizado pelo Programa Escola de Hackers, com o intuito de exemplificar as possibilidades do Scratch para a resolução de problemas matemáticos guiado pelos níveis do domínio cognitivo de Bloom.

Quadro 3 – Sequência didática pautada pelos níveis do domínio cognitivo de Bloom, com atividades do Caderno Didático.

ETAPA	ATIVIDADE
<p style="text-align: center;">1</p> <p style="text-align: center;"> Lembrar e entender</p>	<p>ONDE QUEREMOS CHEGAR? Ao final da desta etapa os estudantes devem recordar, definir, reconhecer ou identificar as potencialidades da programação por blocos a partir do resgate do que os indivíduos aprenderam e vivenciaram anteriormente.</p> <p>ATIVIDADE A SER REALIZADA 1. Criar uma dinâmica de compartilhamento de experiências entre os alunos acerca de seus conhecimentos prévios de programação. Algumas das questões que podem ser feitas para motivar o diálogo são:</p> <ul style="list-style-type: none"> ● O que é programação por blocos? <ul style="list-style-type: none"> ○ Qual o objetivo da programação por blocos? ○ A programação de blocos é de fácil compreensão? ○ Já programaram a partir de blocos? ● Quais softwares são conhecidos pela programação por blocos? ● Na sua opinião, programação por blocos depende de lógica? <p>2. Levando em consideração a atividade anterior, as lembranças e conhecimentos construídos, chegou a hora de identificar o nível de compreensão das informações anteriores por parte dos alunos a partir da apresentação do Scratch. Sugerimos a seguinte sequência de atividades:</p> <ul style="list-style-type: none"> ● Explicação de que trabalharão com Scratch, ambiente de programação em blocos; ● Acesso ao Scratch e apresentação de sua interface.
<p style="text-align: center;">2</p> <p style="text-align: center;"> Aplicar e analisar</p>	<p>ONDE QUEREMOS CHEGAR? Ao final da desta etapa os estudantes devem experimentar o potencial do Scratch como um espaço de programação lúdico e indicado para crianças, bem como das competências cognitivas que o ato de programar exige e desenvolve. Sugere-se realizar esta primeira atividade em duplas ou trios, dependendo do tamanho da turma. É importante que não sejam criados muitos grupos a fim de que se possa proceder às discussões fundamentais para esta aula. O objetivo destas atividades são de possibilitar aos estudantes o aprofundamento do domínio da ferramenta para que possam, em um contexto de ensino de programação, identificar desafios que sejam adequados à faixa etária dos estudantes e passíveis de serem realizados no Scratch.</p> <p>ATIVIDADES A SEREM REALIZADAS Com os questionamentos concluídos e a exploração inicial do Scratch, iremos propor que os alunos solucionem atividades Matemáticas utilizando o Scratch.</p> <p>Atividade 1: MOVE E GIRA Objetivo: Realizar movimentos no sentido horário e anti-horário, além de utilizar-se de ângulos para os giros. Assim como sugere os comandos abaixo:</p> <ul style="list-style-type: none"> ● insira um objeto; ● faça o sprite mover 60 passos para direita, girar 45° no sentido anti-horário e mover outros 60 passos; ● o objeto repete o percurso por 2 vezes; ● desenhar, usando a caneta, a movimentação realizada no palco. <p style="text-align: right;">O desafio resolvido pode ser acessado em: https://scratch.mit.edu/projects/239876073/</p>

<p style="text-align: center;">2  Aplicar e analisar</p>	<p>Atividade 2: OPERAÇÕES MATEMÁTICAS Objetivo: Utilizar operações básicas existentes na Matemática, bem como números antecessores e sucessores. Assim como sugere os comandos abaixo:</p> <ul style="list-style-type: none"> ● faça o sprite perguntar ao usuário que digite um número como entrada; ● calcule o produto desse número com o seu antecessor e seu sucessor; ● faça o gatinho dizer o produto, utilizando uma variável para calcular o produto. <p style="text-align: right;">O desafio resolvido pode ser acessado em: https://scratch.mit.edu/projects/239872496/</p> <p>Atividade 3: VALOR E COMPARAÇÃO Objetivo: Criar um cenário no qual seja possível digitar um valor até 100.000, de tal forma que o personagem escolhido reaja de diferentes formas: se o valor for maior que 5, o sprite deverá dizer “MIAU” e andar 100 passos no palco; Se o valor for igual a 4, o sprite deverá trocar de traje; Se o valor for menor que 4, o sprite deverá dizer o valor digitado, assim como sugere os comandos abaixo:</p> <ul style="list-style-type: none"> ● Pergunte ao usuário que digite um número de entrada; ● Crie uma variável “var” e inicialize seu valor em 1; ● Repita os procedimentos abaixo n vezes, onde n é o número armazenado na variável “resposta” fornecida pelo usuário; ● Use o “Se...então...senão” para verificar se o valor armazenado em “var” é maior que 5, se for faça o gatinho dizer “MIAU” por 2 segundos e andar “var”*100 passos (dica: adicione o comando que permite ao gatinho voltar quando atinge as bordas do cenário). ● Se o valor armazenado em “var” for menor ou igual a 5, faça o gatinho trocar de traje; ● Por fim, adicione o valor de “var” em 1 e faça o gatinho esperar 2 segundos. <p style="text-align: right;">O desafio resolvido pode ser acessado em: https://scratch.mit.edu/projects/239869619/</p>
<p style="text-align: center;">3  Sintetizar</p>	<p>ONDE QUEREMOS CHEGAR? A intenção desta atividade é complexificar as habilidades construídas pelos alunos nas atividades anteriores, relacionando-as com processos de programação de computadores no ensino fundamental. Ao final da atividade os alunos devem estar aptos a construir uma proposta de atividade relacionada com Matemática, a ser executada por crianças em idade escolar utilizando o Scratch.</p> <p>ATIVIDADE A SER REALIZADA</p> <ol style="list-style-type: none"> 1. Propor aos alunos a criação de uma situação Matemática utilizando-se da programação de computadores. A solução deve passar pela construção de uma pequena história, cartão multimídia ou jogo por parte das crianças. Antes de iniciá-la, os estudantes deverão fazer as seguintes definições, denominada de requisitos básicos: <ul style="list-style-type: none"> ● Objetivo do projeto: definição do que os alunos devem construir (cartão multimídia, história ou jogo). ● Caracterização dos alunos: Qual faixa etária e qual ano escolar? ● Que competências desejam desenvolver: Raciocínio lógico? Abstração? Decomposição de problemas? ● Conteúdo a ser trabalhado: Qual o conteúdo a ser tratado no software? 2. Feitas estas definições, cada estudante deverá implementar seu projeto utilizando o Scratch e, com base nisto, disponibilizar o projeto on-line na comunidade do Scratch. 3. Com os projetos concluídos, haverá uma apresentação e comentários sobre os mesmos.

4  Criar	<p>ONDE QUEREMOS CHEGAR?</p> <p>Ao final desta etapa, a turma deve ser capaz de identificar potencialidades e limitações do que foi construído na etapa anterior enquanto proposta de atividade de ensino de programação para crianças da educação básica. A proposta de avaliação deve cumprir os objetivos da sua sequência didática, então relembre que é necessário:</p> <ul style="list-style-type: none"> ● identificar as demandas envolvidas no ensino de programação de computadores para alunos da educação básica; ● reconhecer o potencial da programação de computadores na educação básica, considerando ambientes de programação baseados em blocos. <p>Lembre-se que o mais importante nesta avaliação dos produtos é que sua turma construa conhecimento acerca da ferramenta utilizada, compreendam seu potencial educacional e detenham as bases didáticas mínimas para que possam ensinar programação de computadores para crianças.</p> <p>ATIVIDADE A SER REALIZADA</p> <p>A ideia é que os alunos avaliem, além da real possibilidade de execução da proposta elaborada na Etapa 3, que competências foram melhor demonstradas a partir da análise dos blocos de programação. Para tanto, os alunos devem:</p> <ul style="list-style-type: none"> ● Inserir o seu projeto Scratch no Dr. Scratch e salvar a ficha de avaliação; ● Apresentar seus projetos para a turma, bem como os requisitos básicos definidos; ● Discutir criticamente acerca da pertinência do projeto para os requisitos definidos.
---	---

Fonte: Da autora.

Deste modo, após a exploração deste programa, é possível destacar que a programação de computadores é cada vez mais útil no ambiente educacional, pois pode ser manuseada como instrumento de aprendizagem. Além disso, evidencia-se que o computador enriquece o ambiente de aprendizagem, onde o aluno, interagindo com os objetos desse ambiente, tem a oportunidade de desenvolver habilidades cognitivas.

4 METODOLOGIA E O PRODUTO EDUCACIONAL

Ao oportunizar um espaço diferenciado para o desenvolvimento de competências na área de programação de computadores e raciocínio lógico Matemático no projeto Escola de Hackers, para alunos do Ensino Fundamental II, deseja-se identificar quais os impactos da adaptação do Caderno Didático a partir da Taxonomia de Bloom em relação à primeira versão do mesmo.

Tal Caderno Didático, já existente e aplicado em 2017, foi estruturado com vistas ao conhecimento das diferentes categorias de comandos (*Scripts*¹⁷) contidos no Scratch e exercitar tal conhecimento com a resolução dos problemas e desafios propostos. Desta maneira, o Caderno Didático¹⁸ passou por uma reestruturação, onde foram adicionadas novas atividades a serem executadas pelos alunos e reorganizadas a partir dos seis níveis do domínio cognitivo de Bloom, onde foram ordenados em complexidade crescente, para ser trabalhado no decorrer do ano de 2018, nas cinco escolas participantes do Projeto Escola de Hackers.

Neste contexto, ao tratar-se da Taxonomia de Bloom, assume-se que uma pessoa somente apreende e aplica um determinado conceito depois de inteirar-se do conteúdo ou situação em questão. Tal pressuposto significa que, para adquirir uma nova habilidade que pertence ao próximo nível, um indivíduo precisa ter conquistado e dominado a habilidade do nível anterior.

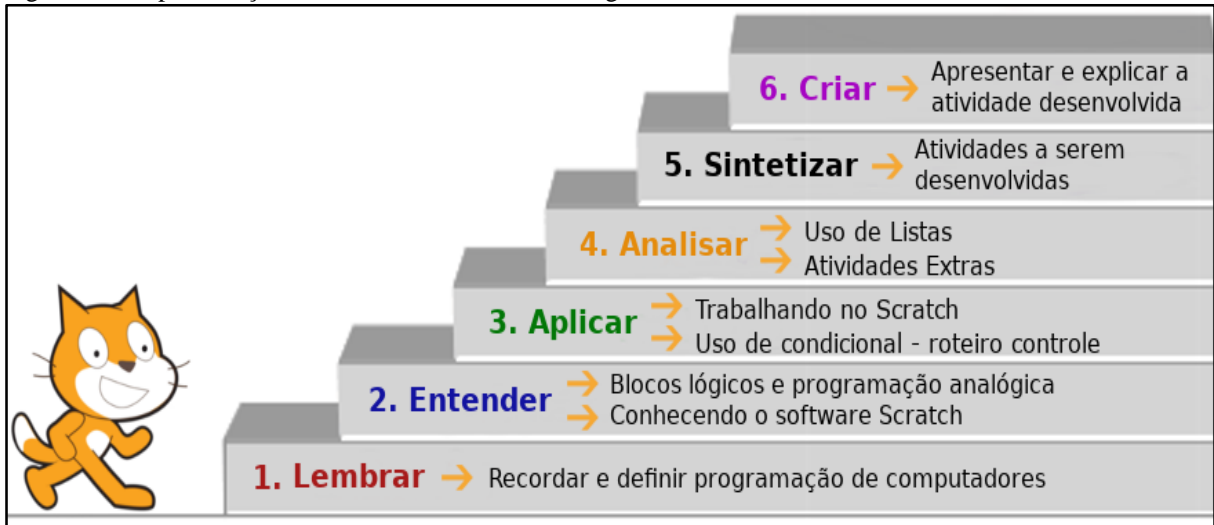
Neste sentido o Caderno Didático se organiza de forma que alunos recordam o que já conheciam de programação de computadores e diante de tal construção, elaboram uma definição para tal termo, passando a conhecer a lógica envolvida neste meio da programação por intermédio dos blocos lógicos, transformando-se em interação com o *software* Scratch.

Foram trabalhadas situações problema e desafios a serem resolvidos dentro da ferramenta, que começou a ser aprofundada utilizando-se de uma nova categoria de comando chamada listas. Diante de um domínio do Scratch, apresentado pelos alunos passou-se a propor atividades extras, jogos e animações mais aprofundados que a etapa anterior a fim de estabelecer relações com a fase antecedente, proporcionando o diálogo entre os alunos e professor, preparando-os para desenvolver o seu próprio jogo, animação, história ou desafio e, ao apresentar, ter a capacidade de explicar onde queriam chegar e porquê. Tal construção está representada na Figura 11.

¹⁷ São ferramentas utilizadas para controlar as ações dos personagens e/ou cenário, de um determinado programa ou aplicativo.

¹⁸ Produto educacional gerado por este estudo, que pode ser acessado através do seguinte link: <https://goo.gl/HNXAJ2>.

Figura 11 - Representação dos seis níveis do domínio cognitivo, conforme estrutura do Caderno Didático.



Fonte: <https://goo.gl/PsFwj>, adaptado pela autora.

Para tanto, além da adaptação do Caderno Didático, os alunos da Escola Municipal de Ensino Fundamental (EMEF) Dom José Gomes, de Passo Fundo, foram observados em suas oficinas semanais no Laboratório Computacional da escola, com duração de 2h, durante duas semanas, período em que a sequência didática proposta no capítulo anterior¹⁹ foi aplicada, buscando dados qualitativos acerca do potencial do Scratch para o desenvolvimento do raciocínio lógico.

Tal sequência didática foi construída com situações problemas contidas no Caderno Didático utilizado no Projeto Escola de Hackers, e foi aplicado primeiramente a um grupo de professores em formação, do curso de Matemática Licenciatura, do Instituto de Ciências Exatas e Geociências, da UPF, a fim de validá-la e receber sugestões para o trabalho que viria a ocorrer com os alunos da EMEF Dom José Gomes.

A coleta de dados, para validação do Caderno Didático adaptado à luz da Taxonomia de Bloom, foi gerada a partir dos mesmos instrumentos utilizados em 2017, ou seja, os estudantes passaram por uma seleção com blocos lógicos em sua escola de origem, responderam o *software* A Prova de Hackers, criado no Scratch, antes de iniciar as atividades no Laboratório Computacional e voltaram a responder no final das ações do Programa. Em seguida, foi realizado a coleta de dados com o *software* Dr. Scratch, utilizando-se de três momentos de cada aluno, ou seja, o primeiro projeto realizado, um intermediário e o último foram acompanhados para obter os desfechos alcançados de cada indivíduo neste ano de

¹⁹ A sequência didática desenvolvida com atividades que envolvem matemática e estão contidas dentro do Caderno Didático, foi descrita neste estudo no capítulo 3, Subseção 3.3.2 (Utilização do Scratch no desenvolvimento de problemas matemáticos), na página 54.

2018. Por fim, foi gerado um comparativo entre os anos 2017 e 2018, para autenticar qual método se apresenta mais eficiente.

Diante do exposto, criamos um blog²⁰ (Anexo A) para hospedar o histórico do Programa Escola de Hackers, Caderno Didático completo do Projeto Escola de Hackers em outra aba, a sequência didática aplicada na seguinte, e os resultados de 2017 e 2018 gerados pelos mesmos meios, isto é, trata-se do teste inicial com blocos lógicos, do *software* A prova de Hackers e a evolução dos alunos identificada dentro da plataforma Dr. Scratch, originando a comparação entre ambos em uma nova aba.

Tal blog (Anexo A) foi desenvolvido, para tornar-se um material de apoio para professores, com o intuito de utilizarem este instrumento dentro das escolas, visando a programação de computadores, com o objetivo de ser referencial, oferecendo suporte necessário para que se sintam encorajados a elaborar suas ações em sala de aula, aproximando conteúdos escolares com a programação de computadores, guiado pelas vivências diárias, visando aprendizado de programação, além de desenvolver competências na área de resolução de problemas, criatividade, raciocínio lógico e trabalho em grupos.

Ao tornar disponível o produto educacional (Caderno Didático), muitos professores de Ensino Fundamental podem utilizá-lo na estruturação de suas aulas de maneira produtiva, com adaptações de acordo com a sua realidade, conteúdo e situação. Desta forma, optou-se por um blog (Anexo A), pois tal plataforma fornece fácil acesso e abre a possibilidade de contar com indicações, sugestões, comentários e críticas dos visitantes.

4.1 Aplicação e análise da sequência didática

A presente proposta de sequência didática buscou aliar a Matemática com a programação de computadores com intuito de estimular o desenvolvimento do raciocínio lógico, visto que, programando os alunos aprendem estratégias importantes para a resolução de problemas. Neste sentido, Dante nos traz que,

devemos propor aos estudantes várias estratégias de resolução de problemas, mostrando-lhes que não existe uma única estratégia, ideal e infalível. Cada problema exige uma determinada estratégia. A resolução de problemas não deve se constituir em experiências repetitivas, através da aplicação dos mesmos problemas (com outros números) resolvidos pelas mesmas estratégias. O interessante é resolver diferentes problemas com uma mesma estratégia e aplicar diferentes estratégias para resolver um mesmo problema. Isso facilitará a ação futura dos alunos diante de um problema novo (2000, p. 59-60).

²⁰ Acesso em: <http://escoladehackersupf.blogspot.com>.

Assim, destaca-se que ao aplicar a sequência didática para a turma de graduação²¹, constatou-se que duas aulas com duração de 4h cada uma é pouco tempo para que o *software* seja bem compreendido, porém os graduandos destacaram a importância de inserir a programação de computadores na escola, para o desenvolvimento do raciocínio lógico, capacidade de resolução de problemas, criatividade, interpretação e concentração.

Após esta fase piloto com a graduação, a mesma sequência didática voltou a ser aplicada em duas aulas de 2h, para cinco alunos entre onze e quatorze anos, sendo duas meninas e três meninos, matriculados no 6º, 7º e 8º ano da EMEF Dom José Gomes, participantes do Projeto Escola de Hackers e ministradas pelo monitor responsável por esta instituição. Por tratar-se de estudantes ligados ao programa, os mesmos já haviam passado pelas duas primeiras etapas da Taxonomia de Bloom (lembrar e entender) e desta maneira, iniciamos a aplicação na terceira fase (aplicar) seguindo até o fim (criar).

Na primeira aula compareceu 80% da turma e seu objetivo foi abordar a segunda fase (aplicar e analisar) da sequência didática²². Foi proposto que os alunos criassem diferentes soluções para as 3 atividades propostas na etapa dois (aplicar e analisar) da sequência didática. Ao final foram questionados sobre as suas construções, buscando explicações das relações estabelecidas para obter a solução.

Desta forma, a primeira atividade proposta tinha o objetivo de realizar movimentos no sentido horário e anti-horário, além de utilizar-se de ângulos para os giros. Os alunos chegaram a um resultado sem dificuldades, pois eles mesmos perceberam que estavam errando ao colocar o *sprite* para andar 60 passos fora do comando repetir, desta forma o personagem estava desenhando uma linha, e assim os estudantes se deram conta que deveria estar tudo contido dentro de repetir para desenhar o quadrado. Logo após, surgiram novas soluções para possíveis alterações de enunciado da atividade, resultando em desenho de um triângulo, pentágono hexágono e círculo.

A segunda tarefa proposta, tinha o objetivo de utilizar-se de operações básicas existentes na Matemática, bem como números antecessores e sucessores. Consequentemente, foi executada sem dificuldades, porém os alunos precisaram ser lembrados o que a palavra “produto” representava e então perceberam que correspondia a uma multiplicação. Desta

²¹ A sequência didática foi aplicada a um grupo de professores em formação, do curso de Matemática Licenciatura, do Instituto de Ciências Exatas e Geociências, da UPF.

²² Aplicar: refere-se à habilidade do aluno em recolher e aplicar informação em situações ou problemas concretos, ou seja, reúne processos nos quais o estudante transporta uma informação genérica para uma situação nova e específica. Analisar: refere-se à habilidade do aluno em estruturar informação, separando as partes das matérias de aprendizagem e estabelecer relações, explicando-as, entre as partes constituintes, ou seja, caracterizam-se por separar uma informação em elementos componentes e estabelecer relações entre eles.

forma, iniciaram o processo de programação de seus *sprites*, para que fizesse uma pergunta (“Insira um número” ou “Digite um número”) e assim, constataram rapidamente que necessitavam utilizar-se do comando variável já que o usuário poderia inserir qualquer número.

Portanto, acrescentaram uma equação com esta variável criada para efetuar a multiplicação entre o algarismo antecessor e sucessor da cifra inserida inicialmente. Em função disto, os alunos testaram a possibilidade de efetuar a adição, subtração e divisão entre o número antecessor e sucessor introduzido premicamente.

A terceira e última atividade era mais complexa, e tinha o objetivo de criar um cenário no qual fosse possível digitar um valor até 100.000, de tal forma que o personagem escolhido reagisse de diferentes formas, conforme o número inserido pelo usuário. Facilmente foi constatado que os alunos estavam apresentando dificuldades e a professora precisou fazer algumas intervenções, que visava esclarecer que o personagem só teria uma reação por vez, dependendo do valor digitado pelo usuário. Desta forma, ajudou os alunos a organizar a programação, buscando o êxito de todos.

Diante do exposto, podemos classificar a primeira aula como produtiva, pois ao final desta etapa percebeu-se que os alunos desenvolveram suas habilidades cognitivas e capacidade de resolução de problemas, pois a todo momento dividiram a atividade proposta em pequenas partes e estabelecem relações com a programação para chegar a um resultado. Enfrentam seus erros como motivação para tentar soluções diferentes, além de propor outras situações, caso o enunciado da atividade fosse modificado, representando novos processos de resolução ao que estavam sugerindo. Assim alcançando o objetivo desta fase.

Na segunda aula também compareceu 80% da turma e abordamos a terceira²³ e quarta²⁴ fase da sequência didática, onde foi proposto aos alunos a criação de uma situação Matemática utilizando-se da programação, assim, os quatro estudantes foram divididos em duplas para originar as suas próprias construções e iniciaram um diálogo com o professor:

Grupo A: *Professora, nós queria criar alguma coisa que envolvesse contas de mais, menos, vezes e dividir, pode ser?*

²³ Sintetizar: refere-se à habilidade do aluno em recolher e relacionar informação de várias fontes, formando um produto novo, ou seja, representa os processos nos quais o estudante reúne elementos de informação para compor algo novo que terá, necessariamente, traços individuais distintivos.

²⁴ Criar: refere-se à habilidade do aluno em fazer julgamentos sobre o valor de algo (produtos, ideias, etc.) tendo em consideração critérios conhecidos, ou seja, representa os processos cognitivos mais complexos. Consiste em confrontar um dado, uma informação, uma teoria, um produto etc... com um critério ou conjunto de critérios, que podem ser internos ao próprio objeto de avaliação, ou externos a ele.

Professora: *Sim, mas vocês não podem fazer nada que seja igual ao que já trabalhamos na outra aula.*

Grupo A: *Pensamos em colocar o personagem perguntando por dois números e depois ele dá a resposta da soma, subtração, multiplicação e divisão... Será que é difícil?*

Professora: *Não pensem na dificuldade... Vocês precisam tentar e eu posso ajudar se for necessário.*

Grupo A: *Nós vamos tentar.*

Minutos depois...

Grupo A: *Professora, já que vamos pedir dois números, que podem ser qualquer um, precisamos criar duas variáveis né?*

Professora: *Sim.*

Minutos depois...

Grupo A: *Professora, você pode nos ajudar a encontrar o erro de programação, porque não está me dando as respostas certas para todas as operações.*

Assim, mudamos pequenos detalhes de ordem ao montar a equação e funcionou o programa criado pelos alunos.

Grupo A: *Estamos muito feliz, professora. Nem estamos acreditando que conseguimos criar um programa.*

Professora: *Parabéns! Eu sempre falo que vocês são capazes e que precisavam tentar.*

Grupo B: *A gente pode pedir para o personagem desenhar uma forma geométrica, tipo um triângulo, onde uma pessoa coloca a medida dos lados e aparece como resposta o perímetro da figura... O que você acha?*

Professora: *Ótima ideia! Tentem executar isso e eu estou a disposição para ajudar caso seja necessário.*

Minutos depois...

Grupo B: *Professora, precisamos de ajuda, porque o nosso triângulo não fecha direito para alguns valores inseridos, mas a medida do perímetro ele responde certo.*

Constatou-se assim, que havia um pequeno problema de angulação na programação, que foi facilmente resolvido.

Grupo B: *A gente também está muito feliz, porque funcionou o nosso programa.*


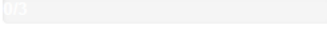




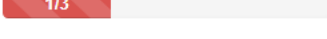

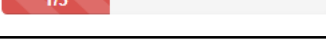
Professora: *Parabéns! Vocês são capazes de fazer outros programas, é só querer.*

Desta forma, atingimos o objetivo desta fase, pois ficou evidente que os alunos estão desenvolvendo suas habilidades de raciocínio lógico, trabalho em grupo e programação de computadores, conseguindo construir uma proposta de atividade relacionada com Matemática, de forma mais complexa que as tarefas propostas em aulas anteriores.

Assim, a professora solicitou que os dois grupos inserissem seus respectivos projetos no Dr.Scratch²⁵, salvassem a ficha de avaliação e ficassem prontos para apresentar suas criações e os resultados obtidos. Ao iniciar as apresentações percebeu-se rapidamente que os alunos compreenderam a lógica de programação, deixando-os seguros para falar o que criaram e o que queriam abordar.

Os alunos do grupo A²⁶, apresentaram testes com diversos números enquanto falavam sobre a sua criação. Constatou-se então, que dominam as quatro operações básicas da Matemática e as funções de programação por blocos, exigida pelo Scratch, criando interatividade com o usuário do seu programa, desenvolvimento do raciocínio lógico, capacidade de abstração e representação de dados. Tais pontos de destaque, também ficaram evidenciado na ficha de avaliação gerada na plataforma Dr. Scratch, como mostra a Figura 12.

Figura 12 - Elementos medidos pelo software Dr. Scratch para o projeto do grupo A, envolvendo as quatro operações básicas da Matemática.

Level up	Nível
 Lógica	
 Paralelismo	
 Interatividade com o usuário	
 Representação de dados	
 Controle de fluxo	
 Sincronização	
 Abstração	

Fonte: Dr. Scratch (<http://www.drscratch.org>).

²⁵ Dr. Scratch é uma ferramenta web gratuita de código aberto, que analisa automaticamente projetos que foram feitos dentro do Scratch. Este dispositivo permite que se descubra um bom programador e o ajuda a atualizar habilidades de codificação. Ao importar um projeto Scratch para o Dr. Scratch, ele fornecerá informações sobre o desenvolvimento do pensamento computacional que o autor obteve, concedendo resultados com valor em relação à capacidade de abstração, pensamento lógico, sincronização, paralelismo, controle de fluxo, interatividade do usuário e representação de dados, podendo atingir a pontuação máxima de 21 pontos, pois cada categoria elencada vale 3 pontos. Acesso disponível no seguinte link: <http://www.drscratch.org/>.

²⁶ O projeto construído por esses alunos pode ser acessado pelo seguinte link: <https://goo.gl/mM8nVb>.

Já os alunos do grupo B²⁷, expuseram diversas medidas para os dois lados do triângulo e testaram a resposta do perímetro no quadro branco, contando com a ajuda dos colegas. verificou-se então, que dominam o conteúdo de geometria básica, abordado no Ensino Fundamental, na disciplina de Matemática e as funções da programação por blocos, exigida pelo Scratch, e assim com a dupla anterior conseguiram gerar interatividade com o usuário do seu programa, capacidade de abstração, representação de dados e desenvolvimento do raciocínio lógico. Mesmo que o *software* não tenha apresentado números reais para esse último, percebe-se que ele está presente pelo domínio matemático apresentado e construção da função que calcula o perímetro dentro do Scratch. Tais pontos de destaque, também ficaram evidenciado na ficha de avaliação gerada na plataforma Dr. Scratch, como mostra a Figura 13.

Figura 13 - Elementos medidos pelo software Dr. Scratch para o projeto do grupo B, envolvendo a geometria básica da Matemática.



Fonte: Dr. Scratch (<http://www.drscratch.org>).

Desta forma, percebe-se que a turma é capaz de identificar potencialidades e limitações de suas construções, conseguindo lidar muito bem com as dificuldades e erros que aparecem no decorrer da elaboração, contribuindo para o seu desenvolvimento do pensamento computacional e do raciocínio lógico, ampliando a capacidade de resolução de problemas, criatividade, curiosidade para novas descobertas, criticidade, reflexão, interpretação, concentração e interação entre os envolvidos.

²⁷ O projeto construído por esses alunos pode ser acessado pelo seguinte link: <https://goo.gl/EaS7MW>.

4.2 Resultados da aplicação do Caderno Didático em 2018

As análises dos resultados²⁸ da aplicação do Caderno Didático reorganizado a luz da Taxonomia de Bloom, em 2018, foram realizadas de forma quantitativa com os mesmos instrumentos utilizados para examinar os dados de 2017, com o objetivo de precisar os resultados que foram avaliados partindo dos frutos colhidos dentro de três frentes diferentes: um teste inicial com blocos lógicos, um *software* produzido dentro do Scratch e a evolução dos alunos identificada dentro da plataforma Dr. Scratch.

O **teste com os blocos lógicos**, pré requisito de integração no projeto, conforme relatado nos resultados obtidos em 2017, foi utilizado para selecionar os alunos das cinco escolas que se prontificaram a participar do programa em 2018. Desta forma, em reunião prévia entre escolas, coordenação do projeto e Prefeitura Municipal de Passo Fundo, acordou-se que seriam selecionados até 15 alunos de cada instituição para envolver-se nesta frente.

Assim, participaram da seleção cerca de duzentos alunos e setenta foram considerados aptos a integrar-se ao Programa Escola de Hackers. Conforme esperado, devido a eficiência deste meio de seleção, os alunos demonstraram-se mais interessados, comprometidos, assíduos e incentivados que em edições anteriores.

O ***software*²⁹ usando o Scratch**, denominado “À Prova de Hackers”, tem como objetivo aferir a capacidade de retenção de informação dos participantes do programa. Assim, foi aplicado no início das atividades e tinha previsão de nova aplicação ao final do ano letivo, como ocorreu em 2017. Tinha como objetivo identificar capacidades cognitivas, em especial o tempo de resposta e a memória de curto prazo dos alunos, para assim ter uma ideia de como o programa estava colaborando para o desenvolvimento dos alunos nestes quesitos.

De acordo com os testes realizados, a média de tempo de todas as questões para as meninas foi de 47 segundos com 69,20% de acertos, e para os meninos a média foi de 51 segundos com 68,10% de acertos. Quando a avaliação voltou a ser aplicada, as meninas diminuíram a média de tempo para 37 segundos e aumentaram seus acertos para 80%, os meninos ampliaram a média de tempo para 54 segundos e melhoraram seus acertos para 80,6%. Portanto, observa-se que as meninas progrediram cerca de 11% nos seus acertos minimizando em 10 segundos o tempo médio e os meninos elevaram o tempo em 3 segundos,

²⁸ A descrição e procedimentos adotados para colher os dados e analisar posteriormente, estão disponíveis no capítulo 2 (Tecnologia como Ferramenta do Ensino), seção 2.5 (Escola de Hackers), subseção 2.5.3 (Resultados do projeto em 2017), página 31 deste estudo.

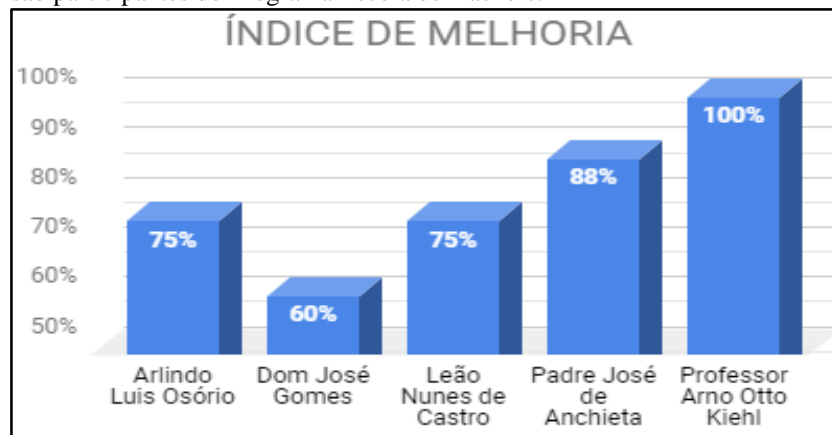
²⁹ O *software* A Prova de Hackers pode ser acessado no Scratch, através do seguinte link: <https://scratch.mit.edu/projects/159196859/>.

porém apresentaram aproximadamente 13% mais acertos. Assim, acredita-se que esses alunos estão expandindo suas aptidões cognitivas, além de um maior desenvolvimento da habilidade de memorização.

Levando em consideração números gerais, detectamos que no primeiro momento os alunos obtiveram uma média de tempo para responder às perguntas de 1 minuto e 17 segundos, com 68,60% de acertos. Ao final do ano, quando o teste voltou a ser aplicado, apuramos que a média de tempo aumentou para 1 minuto e 32 segundos, com 80,30% de acertos. Constatamos assim que 80% dos alunos participantes do programa melhoraram seu tempo de resposta, aumentando significativamente seus acertos nas questões propostas. Tendo em vista os resultados quantitativos, pode-se afirmar que o Programa Escola de Hackers tem auxiliado seus participantes a desenvolver suas capacidades cognitivas, apresentando tempo de resposta mais apurado, além de maior desenvolvimento da habilidade de memorização.

Com a coleta de dados finalizada, os alunos participantes do programa foram divididos em suas respectivas escolas, contendo números totais sobre a melhoria na capacidade de fixação de informação dos alunos. Portanto, entende-se que 80% (vinte e três estudantes) dos participantes ativos neste ano obtiveram melhora significativa na aptidão de retenção de dados, assim como mostra a Figura 14 a seguir.

Figura 14 - Gráfico de apresentação da melhoria na capacidade de retenção de informações dos alunos, representados por suas respectivas escolas, que são participantes do Programa Escola de Hackers.



Fonte: Da autora.

Tratando-se das escolas Arlindo Luis Osório, Dom José Gomes, Padre José de Anchieta e Professor Arno Otto Kiehl, onde ocorreram atividades semanais de 2h, contando com a participação, o apoio e interesse do corpo diretivo e docente de cada instituição para execução do projeto, contribuindo para manter seus alunos interessados, incentivados, participativos e assíduos. Desta forma, detectou-se que dos quatro alunos da Arlindo Luis

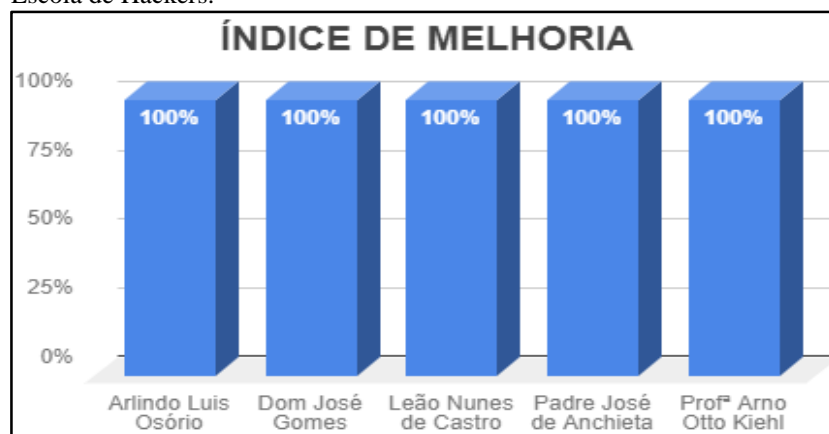
Osório, três melhoraram seu desempenho tanto em tempo como em acertos e um deles manteve a sua performance. A escola Dom José Gomes apresentou melhora de tempo e acertos em três dos cinco alunos participantes, os outros dois mantiveram a quantidade de acertos, apresentando tempo semelhante nos dois momentos. A escola Padre José de Anchieta tinha oito educandos associados ao projeto, assim, exibiu avanço em sete alunos e um desempenho igual no quesito acertos com um tempo equivalente em um estudante. Para a escola Professor Arno Otto Kiehl (quatro participantes) constatou-se que todos os integrantes do programa aperfeiçoaram significativamente sua capacidade de reter informações, onde aumentaram seus acertos em um tempo menor.

Explicitando a escola Leão Nunes de Castro, onde ocorriam oficinas quinzenais de 4h, pois tratava-se de uma instituição do interior e com dificuldade de acesso tanto dos alunos como dos monitores. Também contou com a participação, o apoio e interesse do corpo diretivo e docente para execução do projeto, contribuindo para manter seus alunos interessados, incentivados, participativos e assíduos. Desta maneira, identificamos que dos oito alunos integrantes do programa, seis deles melhoraram significativamente sua capacidade de reter informações, apresentando mais acertos com tempo expressivamente menores.

Já a **plataforma Dr.Scratch**³⁰, analisa automaticamente projetos que foram feitos dentro do Scratch. Esta ferramenta permite que se descubra um bom programador e o ajuda a atualizar habilidades de codificação.

Desta forma, os alunos participantes do programa foram divididos em suas respectivas escolas contendo números totais. Portanto, pode-se afirmar com clareza que os vinte e nove alunos ativos (100%), melhoraram seus índices no decorrer do ano, como mostra a Figura 15.

Figura 15 - Gráfico de aproveitamento das escolas participantes do Programa Escola de Hackers.



Fonte: Da autora.

³⁰ Disponível para acesso em: <http://www.drscratch.org/>.

Considerando as características já citadas das escolas participantes³¹, percebe-se que a contribuição do corpo docente com a proposta e execução do projeto é muito importante para obter bons resultados, assim, evidencia-se que todos os alunos aprimoraram os seus índices no decorrer do ano, desenvolvendo o pensamento computacional, capacidade de resolução de problemas, criatividade, curiosidade, criticidade, reflexão, interpretação, concentração, interação e o raciocínio lógico.

Com os dados levantados a partir das três fontes citadas acima³², é possível afirmar que o programa vem alcançando seus principais objetivos, pois desde a sua concepção busca trazer para seus participantes o desenvolvimento de habilidades que vão além da programação de computadores, ou seja, os alunos devem conseguir desenvolver seu raciocínio lógico, capacidade de memorização e interpretação, ao ponto de obter melhores resultados ao final do ano.

4.2.1 Comparativo dos resultados de 2017 e 2018

Em 2017 havia dez escolas participando do Programa Escola de Hackers, onde o Caderno Didático utilizado estava organizado com situações problema e desafios em ordem crescente de complexidade. Já no ano de 2018 tivemos cinco escolas integrando o projeto, no qual o Caderno Didático passou por uma reorganização à luz da Taxonomia de Bloom, onde foram adicionadas novas atividades a serem executadas pelos alunos e voltou a ser aplicado. Desta forma, as análises dos resultados dos dois anos, foram realizadas de forma quantitativa com os mesmos instrumentos, sendo eles: um teste inicial com blocos lógicos³³, um *software* produzido dentro do Scratch e a evolução dos alunos identificada dentro da plataforma Dr. Scratch.

Tratando-se do ***software produzido dentro do Scratch***, nomeado A Prova de Hackers, percebe-se que os resultados são semelhantes e satisfatórios, conforme a quantidade de alunos que participaram em cada ano. Evidencia-se assim, que a reorganização do Caderno Didático não teve interferência direta com a repercussão final apresentada pelos integrantes do projeto.

Em 2017, envolveram-se ativamente trinta e oito alunos, sendo que trinta e dois deles aperfeiçoaram seus acertos, diminuindo seu tempo para responder às questões, apresentando

³¹ Tais características foram explanadas nas páginas 68 e 69 desta dissertação.

³² Teste inicial com blocos lógicos, software A Prova de Hackers e a plataforma Dr. Scratch.

³³ Pré-requisito para ingressar no Programa Escola de Hackers.

assim, melhoria na capacidade de fixação de conhecimento, totalizando 86% dos estudantes e apenas seis deles mantiveram suas performances iniciais. Em 2018, associaram-se ativamente ao projeto vinte e nove alunos, sendo que vinte e três deles aprimoraram seus acertos, reduzindo seu tempo para solução às questões, expondo assim, refinamento na capacidade de memorização, resultando 80% dos educandos e apenas seis deles conservaram o desempenho inicial, assim como apresenta o Quadro 4.

Quadro 4 - Comparativo de resultados obtidos nos anos de 2017 e 2018 no software A Prova de Hackers pelos alunos vinculados ao Programa Escola de Hackers.

2017				
ESCOLA	Nº DE ALUNOS	MELHOROU	MANTEVE	% DE MELHORA
Arlindo L. Osório	5	4	1	80%
Benoni Rosado	6	6	0	100%
Cohab Secchi - CAIC	3	1	2	33,33%
Dom José Gomes	7	5	2	71,43%
Frederico Ferri	2	2	0	100%
Leão Nunes de Castro	4	4	0	100%
Padre José de Anchieta	5	5	0	100%
Santo Agostinho	1	1	0	100%
Senador Pasqualini	1	1	0	100%
Urbano Ribas	4	3	1	75%
TOTAL	38	32	6	86%
2018				
ESCOLA	Nº DE ALUNOS	MELHOROU	MANTEVE	% DE MELHORA
Arlindo Luis Osório	4	3	1	75%
Dom José Gomes	5	3	2	60%
Leão Nunes de Castro	8	6	2	75%
Padre José de Anchieta	8	7	1	87,50%
Professor Arno Otto Kiehl	4	4	0	100%
TOTAL	29	23	6	80%

Fonte: Da autora.

Abordando a plataforma **Dr. Scratch**, constata-se que os resultados apresentam uma pequena diferença, de acordo com o número de alunos que participou em cada ano do projeto. Desta forma, afirmamos que a reorganização do Caderno Didático à luz da Taxonomia de Bloom interviu para que os estudantes desenvolvessem suas habilidades para programar com mais eficiência.

Em 2017, foi possível selecionar três momentos de cinquenta e oito alunos, sendo que quarenta e seis deles apresentaram evolução no processo de programação no decorrer do ano, totalizando 81% dos estudantes. Em 2018, separamos três produções de vinte e nove educandos, sendo que surpreendentemente todos eles, ou seja, 100% das crianças exibiram

melhora significativa na sequência de programação no decorrer do projeto, conforme o Quadro 5.

Quadro 5 - Comparativo de resultados obtidos nos anos de 2017 e 2018 através da plataforma Dr. Scratch pelos alunos vinculados ao Programa Escola de Hackers.

2017			
ESCOLA	Nº DE ALUNOS	MEHORARAM	% DE MELHORA
Arlindo Luis Osório	6	6	100%
Benoni Rosado	6	6	100%
Cohab Secchi - CAIC	5	5	100%
Dom José Gomes	5	4	80%
Frederico Ferri	6	5	83%
Leão Nunes de Castro	8	6	75%
Padre José de Anchieta	5	5	100%
Santo Agostinho	3	2	67%
Senador Pasqualini	7	0	0%
Urbano Ribas	7	7	100%
TOTAL	58	46	81%
2018			
ESCOLA	Nº DE ALUNOS	MEHORARAM	% DE MELHORA
Arlindo Luis Osório	4	4	100%
Dom José Gomes	5	5	100%
Leão Nunes de Castro	8	8	100%
Padre José de Anchieta	8	8	100%
Profª Arno Otto Kiehl	4	4	100%
TOTAL	29	29	100%

Fonte: Da autora.

Frente aos dados expostos, percebe-se que aprimorar o Caderno Didático levando em consideração a Taxonomia de Bloom, foi eficiente para o ensino na área de programação de computadores, onde os problemas e desafios propostos devem estar organizados em ordem crescente de complexidade, considerando os seis níveis do domínio cognitivo de Bloom, visando o objetivo final de onde quer chegar, além de oportunizar um espaço diferenciado para o desenvolvimento do raciocínio lógico, que considera os erros um ensejo de aprendizagem.

5 CONSIDERAÇÕES FINAIS

A escola deve acompanhar todas as mudanças educacionais e tecnológicas em prol da difusão do conhecimento, desse modo, é notável a necessidade da realização de capacitações para os profissionais da educação, objetivando a conscientização sobre o uso devido das ferramentas tecnológicas, bem como oferecendo suporte a respeito da sua utilização em conjunto com uma metodologia de ensino elaborada especificamente, para o pleno desenvolvimento da prática pedagógica. Utilizando como recurso diversas fontes de objetos de aprendizagem acessíveis às escolas da rede pública, e a rede privada de ensino. Mas para o êxito no processo de informatização da educação é preciso mudar profundamente os métodos de ensino para reservar ao cérebro humano o que lhe é essencial, ou seja, a capacidade de pensar, em vez de apenas memorizar.

O presente estudo teve por finalidade colher resultados do Programa Escola de Hackers, após a reorganização do Caderno Didático, pautado pela Taxonomia de Bloom sobre o processo de ensino de programação, propiciando aos estudantes um espaço diferenciado de desenvolvimento de habilidades que vão além da programação de computadores, visto que os alunos tiveram a oportunidade de desenvolver o raciocínio lógico e o pensamento computacional, a capacidade de memorização, resolução de problemas, criatividade, curiosidade, reflexão, interpretação, concentração e interação.

Percebe-se que quando as escolas abriram suas portas e investiram no ensino de competências digitais, por meio do Programa Escola de Hackers, compreenderam que a tecnologia pode ser utilizada de forma saudável e criativa, desenvolvendo habilidades cognitivas em seus alunos, conforme os resultados obtidos pelo projeto apontaram. Diante deste desfecho buscou-se responder a pergunta de pesquisa: “quais as características e desdobramentos de um processo de ensino de programação de computadores para alunos de Ensino Fundamental para o desenvolvimento do raciocínio lógico?”

Ao destacar as características que estão diretamente ligadas à programação de computadores, detectamos que os alunos tiveram oportunidades de desenvolver a criatividade, procuraram saber como funcionava cada bloco de programação e com isso criaram alternativas para aplicação ao resolver as situações problemas, tomando decisões consistentes, embasadas no conhecimento construído e já fixado, estudando cada caso, e assim demonstrando curiosidade para novas criações. Sentiram-se cada vez mais entusiasmados para seguir em frente no projeto, visto que o erro não era motivo de punição e sim uma

oportunidade de aprendizagem, contribuindo para o aperfeiçoamento da concentração, percepção e estratégia de resolução de problemas.

A criticidade também foi uma habilidade desenvolvida de grande importância, considerando que os alunos apresentaram pensamento crítico construtivo, contribuindo para o processo de reflexão, interpretação e análise do problema, conseqüentemente encontrando uma estratégia de resolução para o mesmo, podendo explicar o caminho utilizado e assim, adquiriram a capacidade de interação com os conhecimentos adquiridos, colegas e professor.

Com relação ao caderno didático e suas duas versões, é possível apontar que, ao levantar os dados no ano de 2017, onde o Caderno Didático utilizado estava organizado com situações problemas e desafios em ordem crescente de complexidade, a partir das três fontes citadas no decorrer do estudo³⁴, foi possível afirmar que o programa estava alcançando seus principais objetivos. Ao coletar os dados do ano de 2018 pelos mesmos meios utilizados no ano anterior, desta vez com o Caderno Didático reorganizado à luz da Taxonomia de Bloom constatamos que a teoria foi ainda mais eficiente para o ensino na área de programação de computadores, oportunizando um espaço diferenciado para o desenvolvimento cognitivo, onde considerou os erros um ensejo de aprendizagem.

É importante ressaltar que a inclusão do ensino de programação nas escolas deve respeitar o desenvolvimento cognitivo e social das crianças. A escola, além de local de aprendizagem, é um espaço onde relações pessoais são construídas através da interação entre os colegas de turma e também com os professores.

Sabendo que o ensino da programação na sociedade atual é de grande valia na formação do indivíduo, é visível que educação e tecnologia são duas frentes que se entrelaçam nesse processo e que aliadas podem provocar mudanças significativas nos ambientes de aprendizagens contemporâneos, trazendo resultados significativos.

Nessa perspectiva, a programação é de grande relevância para o desenvolvimento pessoal e profissional das crianças, dos jovens e adultos, e principalmente para a otimização da educação do país, visto que o ato de ensinar programação nas escolas ajuda nas mais diversas áreas de conhecimento, visto que com a programação de computadores é possível aprender e ensinar os mais diversos assuntos de forma interativa.

³⁴ A coleta de dados foi feita a partir de três fontes diferentes, sendo elas: teste inicial com blocos lógicos, *software* produzido dentro do Scratch e a evolução dos alunos identificada dentro da plataforma Dr. Scratch.

REFERÊNCIAS

- ALMEIDA, Maria Elizabeth Bianconcini de. *Informática e formação de professores*. Secretaria de Educação a Distância. ProInfo, Brasília: Ministério da Educação, SEED, 2000.
- ALVES, Érica Valeria. *As habilidades na solução de problemas em matemática*. 2003. Universidade de Santa Cecília. Disponível em: <<https://goo.gl/pn1q1K>>. Acesso: 10 de março de 2018.
- AZERÊDO, Maria. Alves de. A Matemática no ciclo da alfabetização: aprendendo conceitos numéricos, resolvendo situações-problema. In: FARIA, Evangelina Maria Brito de et al. (Orgs.). *Letramentos em Matemática – PNAIC*. Paraíba. João Pessoa: Editora da UFPB, 2014, v. 1, p. 39-50.
- BLOOM, Benjamin S.; KRATHWOHL, David R.; MASIA, Bertram B. *Taxionomia de Objetivos Educacionais*. Porto Alegre: Editora Globo, 1972.
- BRASIL. Ministério da Educação. *Programa de Matemática do Ensino Básico*. Lisboa: MEDGIDC, 2007. Disponível em: <<https://bit.ly/2Eg2Z2G>> Acesso em: 09 ago. 2018.
- _____. *Formação continuada de professores dos Anos/Séries finais do Ensino Fundamental*. Brasília: 2008.
- _____. *Base Nacional Comum Curricular: educação é a base*. Brasília: 2018. Disponível em: <<https://goo.gl/sQB1xQ>>. Acesso em: 10 out. 2018.
- CORREIA, Luiz Henrique Andrade; SILVA, Alexandre José de Carvalho. *Computador tutelado*. Lavras: Universidade Federal de Lavras - UFLA, 2005.
- DANTE, Luiz Roberto. *Didática da resolução de problemas de matemática*. São Paulo: Ática, 2000.
- DULLIUS, Simone Rosanelli. *O ambiente de autoria Scratch e suas possibilidades de apoio ao processo de aprendizagem*. 2008, 56f. Monografia (Especialização em Informática na Educação) - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2008.
- EDUScratch. Site do Scratch para educadores. Disponível em: <<http://eduscratch.dge.mec.pt/>>. Acesso em: 10 jun. 2017.
- FARIA, Anália Rodrigues de. *O desenvolvimento da criança e do adolescente segundo Piaget*. 4. ed. São Paulo: Editora Ática, 1998.
- FARIA, Eustáquio São José de. *Método trifásico de ensino-aprendizagem baseado na taxionomia de objetivos educacionais de Bloom: uma aplicação no ensino de programações de computadores*. 2010. 295 p. Tese (Doutorado em Engenharia Elétrica) - Universidade Federal de Uberlândia, Uberlândia, 2010.
- GADOTTI, Moacir. *Perspectivas atuais da educação*. Porto Alegre: Artes Médicas, 2000.

JUSTINO, Marinice Natal. *Pesquisa e recursos didáticos na formação e prática docente*. Curitiba: Ibplex, 2011.

MAGER, Robert Frank. *Preparando objetivos para instrução programada*. São Francisco: Fearon, 1961.

MOREIRA, Claudio. *Taxonomia de Bloom: domínios de aprendizagem*. 2009. Disponível em: <<https://goo.gl/QZPLcR>>. Acesso em: 05 mar. 2018.

PAPA, Patricia. *Seymour Papert - o pioneiro da história da informática na educação: matemático, pedagogo e informático*. 2017. Disponível em: <<https://goo.gl/hzzmiJ>>. Acesso em: 18 jul. 2018.

PAPERT, Seymour. *Teaching children thinking (Logo memo)*. Massachusetts Institute of Technology Cambridge, MA, USA, 1971. Disponível em: <<https://goo.gl/9is34j>>. Acesso em: 28 jul. 2017.

_____. *Mindstorms: children, computers, and powerful ideas*. New York: Basic Books, 1980.

_____. *Logo: computadores e educação*. Trad. Editora Brasiliense S. A. São Paulo: Brasiliense S. A, 1985.

_____. *A máquina das crianças: repensando a escola na era da informática*. Porto Alegre: Artmed, 2008.

PEREIRA, João Thomaz. Educação e sociedade da informação. In: COSCARELLI, Carla; RIBEIRO, Ana Elisa (Orgs.). *Letramento digital: aspectos sociais e possibilidades pedagógicas*. Belo Horizonte: Ceale; Autêntica, 2005. p. 13-24.

PIAGET, Jean. *O nascimento da inteligência na criança*. 3. ed. Rio de Janeiro: Zahar, 1978.

PÓLYA, George. *A arte de resolver problemas*. Rio de Janeiro: Interciência, 2003.

_____. *Como resolver problemas*. Lisboa: Gradiva, 2003.

PONTE, João Pedro da; BROCARD, Joana; OLIVEIRA, Hélia. *Investigações matemáticas na sala de aula*. v. 7. Belo Horizonte: Autêntica, 2007. (Coleção Tendências em Educação Matemática).

RAMPANELLI, Marília et al. Projeto Mutirão pela Inclusão Digital: relato de uma experiência de programação de computadores na Educação Infantil. In: SIMPÓSIO INTERNACIONAL DE INFORMÁTICA EDUCATIVA, 17, 2015, Setúbal, Portugal. *Anais...* Setúbal, Portugal: Escola Superior de Educação do Instituto Politécnico de Setúbal, 2015. p. 234-240. Disponível em: <<http://siie15.ese.ips.pt/>>. Acesso em: 09 ago. 2018.

RESNICK, Mitchel; ROSENBAUM, Eric. Designing for tinkerability. In: HONEY, Margaret; KANTER, David E. (Eds.). *Design, make, play: growing the next generation of STEM innovators*. Londres: Routledge, 2013. p. 163-181.

RUSHKOFF, Douglas. *As 10 questões essenciais da era digital*. São Paulo: Editora Saraiva, 2012.

SCOLARI, Angélica Taschetto; BERNARDI, Giliane; CORDENONSI, Andre Zanki. O desenvolvimento do raciocínio lógico através de objetos de aprendizagem. *Revista Renote – Novas Tecnologias na Educação*, Porto Alegre, v. 5, n. 2, p. 1-10, 2007.

SCRATCH. About Scratch (Scratch Documentation Site). Disponível em: <<https://goo.gl/jQo3AU>>. Acesso 10 jun. 2017.

VALENTE, José Armando. *A telepresença na formação de professores da Área de Informática em Educação: implantando o construcionismo contextualizado*. In: CONGRESSO RIBIE, 4, 1998, Brasília. *Anais...* Brasília: NIED-UNICAMP, 1998. Disponível em: <<https://goo.gl/Py4EbY>>. Acesso em: 17 out. 2017.

VYGOTSKY, Lev Semyonovich. *A formação social da mente*. 6. ed. São Paulo: Martins Fontes, 1989.

WING, Jeannette M. *Computational thinking: it represents a universally applicable attitude and skill set everyone, not just computer scientists, would be eager to learn and use*. *Communications of The ACM*, v. 49, n. 3, p. 33-35, mar. 2006.

ANEXO A - Exposição do Blog

O blog criado tem o objetivo de ser referencial e suporte necessário para que professores se sintam encorajados a elaborar suas ações em sala de aula, aproximando conteúdos escolares com a programação de computadores, guiado pelas vivências diárias, visando aprendizado de programação, além de desenvolver competências na área de resolução de problemas, criatividade, raciocínio lógico e trabalho em grupos. Espera-se que muitos professores de Ensino Fundamental, possam utilizá-lo na estruturação de suas aulas de maneira produtiva, com adaptações de acordo com a sua realidade, conteúdo e situação.

O blog em questão³⁵, se exhibe na forma de uma página na Web, que é alimentada e atualizada frequentemente, composto por postagens sobre o Projeto Escola de Hackers, Programação de Computadores, *software* Scratch e os meios utilizados para validar o caderno didático. Apresentam-se de forma cronológica, seguindo uma linha de tempo com uma postagem após a outra, em que a publicação mais recente fica sempre no topo da página. Essas postagens podem ser escritas pelo autor do blog ou por uma lista de membros que ele convida e autoriza a postar mensagens.

Desta forma, tal blog está estruturado com uma página inicial (Figura 1), que contém a descrição e as abas com conteúdo do mesmo, sendo elas: O Programa, Caderno Didático, Sequência Didática e Resultados. Logo abaixo apresentam-se as publicações mais recentes e no canto superior esquerdo aparecem os colaboradores do blog. Algumas abas possuem extensões, no caso do programa, contém o histórico, objetivos e metodologia utilizada no projeto e a premiação recebida recentemente. Já no caso dos resultados, inclui números obtidos no ano de 2017 e 2018, após o seu comparativo nos três meios de avaliação, que são: Seleção dos alunos com blocos lógicos, um *software* produzido dentro do Scratch e a evolução dos alunos identificada dentro da plataforma Dr. Scratch.

³⁵ Disponível para acesso em: <http://escoladehackersupf.blogspot.com/>.

Figura 1- Página inicial do blog.



Fonte: Blog Escola de Hackers.

A página denominada “O Programa” (Figura 2) apresenta uma breve explicação do que é o Projeto Escola de Hackers, sendo assim esclarece que trata-se de um programa realizado pela Prefeitura Municipal de Passo Fundo, que oportuniza um espaço de aprendizagem de programação de computadores para estudantes do Ensino Fundamental da rede pública, usando o *software* Scratch. Neste sentido, este menu objetiva apresentar e detalhar o programa Escola de Hackers em todas as suas frentes, podendo acessar histórico, objetivos, metodologia e a premiação recebida.

Figura 2 - Página “O Programa” do blog.



Fonte: Blog Escola de Hackers.

A página nomeada “Caderno Didático” (Figura 3) expõe todo material proposto nas oficinas semanais que ocorrem nas escolas participantes do programa e tem o intuito de guiar o professor no seu planejamento, conforme a sua realidade e conteúdo trabalhado. Ele é estruturado em resolução de problemas e desafios. Desta forma, inicia abordando a lógica com o auxílio dos blocos lógicos e interação entre participantes, em seguida, passa à apresentação do *software* Scratch, utilizado pelo projeto e após inicia as atividades referentes à programação de computadores, organizadas em ordem crescente de complexidade.

Figura 3 - Página “Caderno Didático” do blog.

Postado por Márcia Corsetti

CADERNO DIDÁTICO

• **Atividade 10 - JUNTE**

- ❑ Crie duas variáveis, chamadas PRIMEIRA e SEGUNDA;
- ❑ Faça o gatinho perguntar ao usuário que informe dois valores e atribua-os um em cada variável;
- ❑ A seguir, troque os valores, de forma que PRIMEIRA contenha o valor de SEGUNDA e vice-versa.
- ❑ O gatinho deve repetir os novos valores.

Figura: Sugestão de comando para execução da atividade 10.

• **Funções e Operações Matemáticas**

• **Atividade 11 - PRODUTO**

- ❑ Faça o ator perguntar ao usuário que digite um número como entrada, e grave a resposta em uma variável;
- ❑ Calcular o produto de seu antecessor com o seu sucessor, $(n-1) \cdot (n+1)$;
- ❑ Faça o gatinho (ator) dizer o resultado.

Fonte: Blog Escola de Hackers.

A página designada “Sequência Didática” (Figura 4) exhibe uma sucessão de exercícios matemáticos organizados dentro da Taxonomia de Bloom, a fim de apresentar como a programação de computadores pode ser abordada em sala de aula pelo professor, facilitando seu planejamento. Trata-se de atividades que estão contidas no caderno didático e podem ser modificadas conforme o conhecimento dos alunos em relação ao *software* utilizado e conteúdo abordado.

Figura 4 - Página “Sequência Didática” do blog.

SEQUÊNCIA DIDÁTICA

atividades realizadas para que possam ensinar programação de computadores para crianças.

ATIVIDADE A SER REALIZADA

A ideia é que os alunos avaliem, além da real possibilidade de execução da proposta elaborada na Etapa 3, que competências foram melhor demonstradas a partir da análise dos blocos de programação. Para tanto, os alunos devem:

- Subir seu projeto no Dr. Scratch e salvar a ficha de avaliação - figura 1.
- Apresentar seus projetos para a turma, bem como os requisitos básicos definidos.
- Discutir criticamente acerca da pertinência do projeto para os requisitos definidos;

Figura 1: Tabela de avaliação que o Software Dr. Scratch apresenta.

Level up	Nível
Lógica	100
Paralelismo	100
Interatividade com o usuário	75
Representação de dados	75
Controle de Fluxo	100
Sincronização	100
Abstração	100

Deste modo, após a exploração deste programa, é possível destacar que a programação de computadores é cada vez mais útil no ambiente educacional, pois pode ser manuseada como instrumento de aprendizagem. Além disso, evidencia-se que o computador enriquece o ambiente de aprendizagem, onde o aluno, interagindo com os objetos desse ambiente, tem a oportunidade de desenvolver habilidades cognitivas.

Fonte: Blog Escola de Hackers.

A página intitulada “Resultados” (Figura 5) revela os meios utilizados para coleta de dados a fim de analisar os resultados obtidos com a aplicação do caderno didático, este estudo foi realizado de forma quantitativa com o objetivo de precisar os resultados que foram avaliados partindo dos dados gerados dentro de três instrumentos diferentes: um teste inicial com blocos lógicos, um *software* produzido dentro do Scratch nomeado como “A Prova de Hackers” e a evolução dos alunos identificada dentro da plataforma Dr. Scratch. Desta forma, os professores têm a possibilidade de replicar os instrumentos de coleta de dados.

Figura 5 - Página “Resultados” do blog.

Este blog tem o objetivo de ser referencial para professores, oferecendo suporte necessário para que se sintam encorajados a elaborar suas ações em sala de aula, aproximando conteúdos escolares com a programação de computadores, guiado pelas vivências diárias, visando aprendizado de programação, além de desenvolver competências na área de resolução de problemas, criatividade, raciocínio lógico e trabalho em grupos.

[Página Inicial](#) [Programa](#) [Caderno Didático](#) [Sequência Didática](#) [Resultados](#)

Elaborado por Milene Clemente

RESULTADOS

ESCOLA DE HACKERS Este blog é o produto educacional de um estado feito dentro do Programa de Pós Graduação em Ensino de Ciências e Matemática, da Universidade de Passo Fundo e será desenvolvido na forma de material de apoio para professores utilizarem este instrumento dentro das escolas, visando a programação de computadores, e irá referir-se ao caderno didático elaborado para o estudo. Tal caderno didático, estará apoiado na Taxonomia de Bloom, entrando em contato direto com as ideias de Seymour Papert.

O objetivo deste material é ser referencial e suporte necessário para que professores se sintam encorajados a elaborar suas ações em sala de aula, aproximando conteúdos escolares com a programação de computadores, guiado pelas vivências diárias, visando aprendizado de programação, além de desenvolver competências na área de resolução de problemas, criatividade, raciocínio lógico e trabalho em grupos. Espera-se que, muitos professores de Ensino Fundamental, possam utilizá-lo na estruturação de suas aulas de maneira produtiva, com adaptações de acordo com a sua realidade, conteúdo e situação. Para tanto, as análises dos resultados da aplicação do caderno didático serão realizadas de forma quantitativa com o objetivo de precisar os resultados, que serão avaliados partindo dos frutos colhidos dentro de três frentes diferentes, sejam elas: Um [software](#) produzido dentro do Scratch, um teste com blocos lógicos e a evolução dos alunos identificada dentro da plataforma [DeScratch](#).



- 2017
- 2018
- Comparativo

Fonte: Blog Escola de Hackers.

O blog está disponível para acesso, fornecendo uma ferramenta de fácil entrada que possibilita contar com indicações, sugestões, comentários e críticas dos visitantes. Desta maneira, está estruturado de forma a que muitos docentes de Ensino Fundamental possam utilizá-lo na estruturação de suas aulas com aspecto atrativo e produtivo.

PRODUTO EDUCACIONAL

O Produto Educacional encontra-se disponível nos endereços:

<http://docs.upf.br/download/ppgecm/Milene_PRODUTO.pdf>

<<http://educapes.capes.gov.br/handle/capes/431416>>



Milene Giaretta

CADERNO DIDÁTICO

Produto Educacional apresentado ao Programa de Pós-Graduação em Ensino de Ciências e Matemática, do Instituto de Ciências Exatas e Geociências, da Universidade de Passo Fundo, como requisito parcial para a obtenção do grau de Mestre em Ensino de Ciências e Matemática, sob a orientação do Professor Dr. Adriano Canabarro Teixeira.

Passo Fundo
2018

CIP – Catalogação na Publicação

G435p Giaretta, Milene

O programa Escola de Hackers e sua contribuição para o desenvolvimento do raciocínio lógico em crianças do Ensino Fundamental II [recurso eletrônico] / Milene Giaretta. – 2018.

3.2 Mb ; PDF. – (Produtos Educacionais do PPGECEM).

Inclui bibliografia.

ISSN 2595-3672

Modo de acesso gratuito: <<http://www.upf.br/ppgecem>>.

Este material integra os estudos desenvolvidos junto ao Programa de Pós-Graduação em Ensino de Ciências e Matemática (PPGECEM), na Universidade de Passo Fundo (UPF), sob orientação do Prof. Dr. Adriano Canabarro Teixeira.

1. Programação (Computadores) - Estudo e ensino. 2. Lógica no ensino. 3. Ensino Fundamental. I. Teixeira, Adriano Canabarro, orientador. II. Título.

CDU: 37:004

Catálogo: Bibliotecário Luís Diego Dias de S. da Silva – CRB 10/2241

APRESENTAÇÃO

Este produto educacional é um material de apoio pensado e destinado a professores de Ensino Fundamental, que desejam incorporar a programação de computadores no ensino de sua disciplina, referindo-se a utilização dos recursos do software Scratch para a programação. Trata-se de uma proposta didática de atividades, desenvolvida junto ao Programa Escola de Hackers e ao Programa de Pós-Graduação em Ensino de Ciências e Matemática, vinculado a linha de pesquisa tecnologias de informação, comunicação e interação aplicadas ao ensino de Ciências e Matemática, da Universidade de Passo Fundo, sob a orientação do professor Dr. Adriano Canabarro Teixeira.

Embora sejam inúmeras as iniciativas destinadas ao ensino de programação na educação formal, algumas merecem destaque pela forma como são organizadas e pela capilaridade de suas ações. Dentre elas, destaca-se o Programa Escola de Hackers realizado pela Prefeitura Municipal de Passo Fundo e que, desde 2014, já ensinou programação de computadores para mais de 1500 crianças e adolescentes da rede municipal de ensino. Marcado por metodologia própria que visa o desenvolvimento de competências, mais do que habilidade de programação por si só, tem sido campo de pesquisas na área de Educação e de Ensino.

O Projeto Escola de Hackers ocorre no laboratório computacional de cada escola participante do programa, com oficinas semanais ou quinzenais, com duração de 2h ou 4h respectivamente, ministradas por um monitor. Neste sentido, todas as atividades trabalhadas nas aulas foram ordenadas e disponibilizadas neste caderno didático, organizando-se de forma a criar situações problema onde, além de refletir sobre as melhores formas de resolução, criam oportunidades de aprendizado e exercício de competências na área de programação de computadores.

O material tem o objetivo de ser referencial e suporte necessário para que professores se sintam encorajados a elaborar suas ações em sala de aula, aproximando conteúdos escolares com a programação de computadores, guiado pelas vivências diárias, visando aprendizado de programação, além de desenvolver competências na área de resolução de problemas, criatividade, raciocínio lógico e trabalho em grupos. Espera-se que muitos professores de Ensino Fundamental, possam utilizá-lo na estruturação de suas aulas de maneira produtiva, com adaptações de acordo com a sua realidade, conteúdo e situação.

Aprender a programar fornece inúmeros benefícios, tanto para os estudantes como para as escolas que lecionam essa habilidade. Ao programar, o aluno lida com vários desafios

mentais, que o forcem a pensar de forma abstrata, sequencial e organizada, além de proporcionar ao aluno um universo desafiador que o desperta para pensar. Isso implica em um sistema de aprendizagem ativa, onde a composição do conhecimento se dá a partir das ações físicas ou mentais do aluno.

Desta forma, ao potencializar o ensino de programação nas escolas é incentivar professores e estudantes a criar aplicativos para a resolução de problemas sociais que envolvam a comunidade onde vivem. São ideias simples e inovadoras de como abordar o uso da tecnologia no processo de ensino-aprendizagem de uma forma mais dinâmica e interativa, que poderiam contemplar a atual proposta de Base Nacional Comum Curricular, especialmente quando trata do desenvolvimento do pensamento computacional, tendo em vista a capacidade dos alunos de interpretar situações problemas, investigar, decompor em partes mais simples e ordená-las conforme a construção mental, para obter seus resultados através da programação de computadores, assim estarão aptos a explicá-los de maneira clara e argumentos concretos.

Com as mudanças tecnológicas e informacionais que a sociedade sofreu nos últimos anos, é cada vez mais necessário que as instituições de ensino incorporem à sua grade consolidada novas matérias que atendam as demandas do século XXI, sendo o ensino de programação nas escolas uma das mais importantes.

COLABORADORES DO CADERNO DIDÁTICO

Adriano Canabarro Teixeira (Bacharelado em Ciência da Computação - UPF)
Andreza Moreira Santarém (Bacharelado em Ciência da Computação - UPF)
Ariane Mileidi Pazinato (Licenciatura em Matemática - UPF)
Diélisson Geleski (Curso Superior de Análise e Desenvolvimento de Sistemas – UPF)
Débora Sernajotto (Licenciatura em Matemática - UPF)
Evandro Luís Viapiana (Bacharelado em Ciência da Computação - UPF)
Gabriela Colussi (Bacharelado em Psicologia - UPF)
Guilherme Alvim Barbosa (Licenciatura em Física - UPF)
Gustavo de Oliveira da Rosa (Licenciatura em Matemática - UPF)
Jader Hartmann (Bacharelado em Ciência da Computação - UPF)
Jonas Vaz (Bacharelado em Ciência da Computação - UPF)
Jordano Martins Fritsch (Bacharelado em Ciência da Computação - UPF)
Jônatas Strapazon (Bacharelado em Ciência da Computação - UPF)
Kimberly Rolim (Bacharelado em Ciência da Computação - IFRS)
Lucas Vieira May (Curso Superior de Tecnologia em Sistema para Internet - Imed)
Marília Rampanelli (Licenciatura em Matemática - UPF)
Milene Giaretta (Licenciatura em Matemática - UPF)
Neuza Terezinha Oro (Licenciatura Plena em Matemática - UPF)
Pâmela Luiza Lauxen (Licenciatura em Pedagogia - UPF)
Ricardo Matheus Fachinello (Bacharelado em Ciência da Computação - UPF)
Tháise Uczay de Jesus (Licenciatura em Matemática - UPF)

SUMÁRIO

1	BLOCOS LÓGICOS E PROGRAMAÇÃO ANALÓGICA.....	8
1.1	Atividades com os Blocos Lógicos	8
1.2	Atividades de programação analógica	12
2	CONHECENDO O SOFTWARE SCRATCH.....	14
2.1	Tela Inicial.....	14
2.2	Palco	14
2.3	Atores	17
2.3.1	<i>Atividade 1</i>	23
2.3.2	<i>Atividade 2</i>	24
2.3.3	<i>Atividade 3</i>	24
2.3.4	<i>Atividade 4</i>	24
2.3.5	<i>Atividade 5</i>	25
2.4	Roteiros	25
2.4.1	<i>Roteiros de movimento</i>	25
2.4.2	<i>Roteiros de controle.....</i>	26
2.4.3	<i>Roteiros relativos à eventos.....</i>	27
2.4.4	<i>Roteiros relativos à aparência.....</i>	27
2.4.5	<i>Roteiros que se constituem em sensores</i>	28
2.4.6	<i>Roteiros para manipulação de som</i>	29
2.4.7	<i>Roteiros que implementam operadores</i>	30
2.4.8	<i>Roteiros para uso da caneta</i>	30
2.4.9	<i>Roteiros para utilização de variáveis</i>	31
3	TRABALHANDO NO SCRATCH	33
3.1	Atividade 1 - ATOR E PALCO	33
3.2	Atividade 2 - CENTRO DE ROTAÇÃO	34
3.3	Atividade 3 - MOVIMENTO DO ATOR USANDO AS SETAS.....	35
3.4	Atividade 4 - LOTERIA	36
3.5	Atividade 5 - DESENHAR INICIAL	37
3.6	Atividade 6 - MAIS BLOCOS	38
4	USO DE CONDICIONAL - ROTEIRO CONTROLE	40
4.1	Atividade 7 - GOLEIRA	40
4.2	Atividade 8 - VARIÁVEL E RESPOSTA	42

4.3	Atividade 9 - VARIÁVEL DE CONTAGEM	43
4.4	Atividade 10 - JUNTE	44
4.5	Atividade 11 - PRODUTO	45
4.6	Atividade 12 - MÉDIA ARITMÉTICA	46
5	USO DE LISTAS	47
5.1	Atividade 13 - LISTA DE MATÉRIAS	47
5.2	Atividade 14 - LISTA DE NÚMEROS	48
5.3	Atividade 15 - INVERTENDO OS ELEMENTOS.....	49
5.4	Atividade 16 - NOME E IDADE	49
5.5	Atividade 17 - LOTERIA DE PALAVRAS.....	50
5.6	Atividade 18 - DE TRÁS PARA FRENTE.....	51
5.7	Atividade 19 - CRIE CLONES.....	54
5.8	Atividade 20 - PEGUE A MAÇÃ	55
5.9	DESAFIO 01.....	56
5.10	Atividade 21 - FURAR BALÃO	58
5.11	DESAFIO 02.....	60
5.12	Atividade 24 - FOGOS DE ARTIFÍCIO	61
5.13	DESAFIO 03.....	63
5.14	Atividade 25 - TYPE.....	64
5.15	DESAFIO 04.....	65
5.16	Atividade 26 - MÚSICA	66
5.17	Atividade 27 - PARTITURAS	67
5.18	DESAFIO 05.....	68
6	ATIVIDADES EXTRAS	72
6.1	Atividade 1 - CRIAR UM ATOR.....	72
6.2	Atividade 2 - INSERIR SOM NO ATOR.....	72
6.3	Atividade 3 - MUDAR A FANTASIA.....	72
6.4	Atividade 4 - IMPORTAR ATORES.....	72
6.5	Atividade 5 - PERGUNTA	73
6.6	Atividade 6 - VARIÁVEL RESPOSTA.....	73
6.7	Atividade 7 - TAMANHO DO ATOR	73
6.8	Atividade 8 - RAIZ QUADRADA	74
6.9	Atividade 9 - DIA E NOITE	74
6.10	Atividade 10 - TAMANHO DA PALAVRA.....	75

6.11	Atividade 11 - ADIÇÃO DE VARIÁVEIS	75
6.12	Atividade 12 - LABIRINTO	76
6.13	Atividade 13 - MULTIPLICAÇÃO DE VARIÁVEIS	77
6.14	Atividade 14 - MATEMÁTICA DIVERTIDA	77
6.15	Atividade 15 - VOCABULÁRIO	78
6.16	Atividade 16 - ANÃO OU GIGANTE.....	78
6.17	Atividade 17 - TOCA E DANÇA.....	78
6.18	Atividade 18 - CARIMBANDO O TRAJE.....	79
6.19	Atividade 19 - MAIOR, MENOR E IGUAL	79
6.20	Atividade 20 - ASSINATURA	80
6.21	Atividade 21 - ADIÇÃO COM O USO DE LISTA	81
7	ATIVIDADES A SEREM DESENVOLVIDAS.....	82
	REFERÊNCIAS	84

1 BLOCOS LÓGICOS E PROGRAMAÇÃO ANALÓGICA

As atividades apresentadas a seguir foram desenvolvidas no Projeto Escola de Hackers com o intuito de que os alunos participantes compreendessem a lógica dos comandos utilizados na programação de computadores. Elas envolveram os blocos lógicos e encenação de rotinas utilizadas em programação, que chamaremos de programação analógica.

1.1 Atividades com os Blocos Lógicos

- Iniciar as atividades com a apresentação do monitor e dos alunos,
- Estabelecer um contrato didático: regras de boa convivência para o andamento das aulas (conversar com os alunos sobre estas regras),
- Apresentar os blocos lógicos aos alunos,
- Classificá-los quanto aos atributos: cor, tamanho, forma e espessura.
- Em seguida trabalhar com os alunos sobre os Blocos Lógicos, da seguinte maneira:

a) PERGUNTAR AOS ALUNOS:

- Vocês conhecem os blocos lógicos?
- O que vocês entendem por blocos lógicos?
- O que vocês lembram de blocos lógicos?

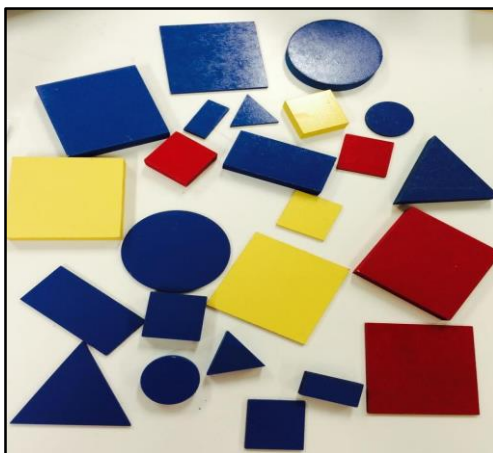
b) Dividir os alunos em grupos,

c) O professor apresenta alguns exemplos, usando sentenças que envolva

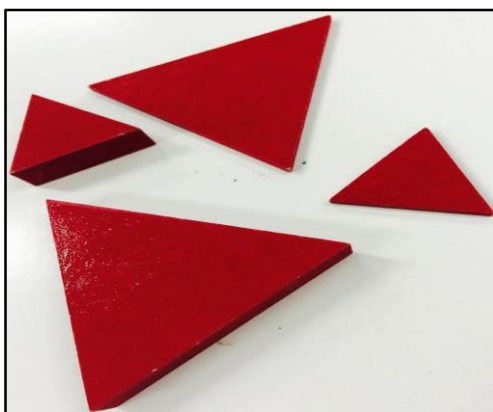
- O conectivo “e”: separe blocos amarelos e circulares (figura)



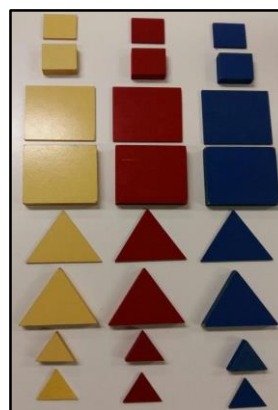
- o conectivo “ou”: separe blocos azuis ou quadrados (figura)



- condicional “se ... então”: Se blocos vermelhos, então triangulares.



- Condicional “se... então, senão”: Se blocos circulares, então coloridos. Senão triângulos e quadrados coloridos



d) Dividir os alunos em grupo e solicitar separem os blocos lógicos conforme as sentenças:

- Blocos azuis e quadrados;
- Blocos azuis ou quadrados;
- Blocos circulares e grandes;
- Blocos amarelos ou grossos;
- Blocos finos e vermelhos;
- Blocos circulares ou grandes;

- Se blocos azuis, então quadrados;
- Se blocos triangulares, então coloridos;
- Se blocos finos, então retângulos;
- Se blocos azuis, então quadrados. Senão, círculos, triângulos e retângulos.
- Se triângulos ou círculos, então colorido. Senão, quadrados e retângulos.
- Se blocos grossos, então coloridos. Senão, finos.

e) Dividir os alunos em grupos (sugerimos 2 ou 3 alunos por grupo).

Solicitar que cada grupo elabore sentenças usando e, ou, se... então e se... então, senão. Após, cada grupo aplica ao outro grupo participante da atividade (a escolha do grupo que irá responder a (as) sentença(s) fica a critério dos participantes da atividade)

f) Jogo do Pirata

A atividade tem por objetivo estimular a comparação visual e exercitar a comparação entre os atributos, para que compreendam que a negação leva à classificação e ajuda a compreender, por exemplo, que um número pertence a um e não a outro conjunto.

Primeiramente o professor contará a história do Pirata e logo após irá estimular uma caça ao tesouro. Desta forma o professor (capitão) imaginará uma peça dos blocos lógicos (tesouro) e dará as dicas até encontrar o aluno (marujo) que tem a peça de blocos lógicos (tesouro) imaginada pelo capitão, para que o tesouro seja descoberto.

A atividade começa com a contação da história: “Era uma vez, um pirata que adorava tesouros. Havia, no porão de seu navio, um baú carregado de pedras preciosas. Nesse porão, ninguém entrava. Somente o pirata tinha a chave. Mas sua felicidade durou pouco. Numa das viagens, uma tempestade virou seu barco e obrigou todos os marinheiros a se refugiarem numa ilha. Furioso, o pirata ordenou que eles voltassem a nado para resgatar o tesouro”.



A história será a chave para descobrir o “marujo” que está com o tesouro. Novamente em grupos:

- Escolha do capitão;
- Entrega de um Bloco Lógico por grupo;
- Exceto o capitão, os demais integrantes do grupo (marujos) escolhem as peças dos blocos de tal forma de que todas fiquem distribuídas (o número de peças não precisa ser igual para todos os integrantes do grupo);

- O capitão dá dicas da peça usando os atributos dos Blocos Lógicos (Diz sentenças lógicas);
- Os marujos observam em suas peças as atribuições enunciadas, descartando as peças que não satisfazem a sentença dita pelo capitão;
- O tesouro será a peça que estará com o marujo que ao final estiver com uma única peça.



Supondo que a peça escolhida seja um triângulo, pequeno, azul e grosso, você diz: “**Quem pegou o tesouro tem a peça azul**”. Em seguida, dê outra dica: “**Quem pegou o tesouro tem a forma triangular**”. Siga até chegar ao marujo que esconde o tesouro.

g) Dividir a turma em 3 grupos, por tributo cor.

- Distribuir as peças separadas por cor, entre os integrantes de cada grupo;
- A criança que tiver pelo menos uma das peças que obedeça a sentença sorteada, executa a ação;
- Ganha a equipe cuja cor chegar ao fim primeiro.

VERMELHO	AMARELO	AZUL
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	10	10

- Questões que serão sorteadas e lidas pelo professor:
 - Se blocos grossos, então ande uma casa;

- Se blocos pequenos ou amarelos, então ande uma casa;
- Se triângulos e finos, então ande duas casas;
- Se grandes e grossos, então: se estiver na largada, permaneça na sua posição; senão volte uma casa;
- Se grossos, grandes e amarelos, então ande duas casas;
- Se blocos finos, então retorne uma casa;
- Se não amarelo, grandes e grossos, então ande uma casa;
- Se grandes, finos e azuis, então ande três casas;
- Se quadrado e pequeno, então ande uma casa;
- Se círculo grande, fino e não azul, então ande duas casas;
- Se blocos não grossos e não grandes, então ande quatro casas;
- Se retângulos grandes e grossos, então ande três casas;
- Se círculo grande e fino, então ande duas casas;
- Se blocos grossos e pequenos, então ande cinco casas;
- Se círculo fino, então volte uma casa;
- Se blocos grossos e amarelo, então ande duas casas;

1.2 Atividades de programação analógica

a) Questionar os alunos: O que é programar? (Entende-se que programar é dar uma série de comandos para alguém/algo executar).

b) Convidar um aluno para que se aproxime do professor na frente da turma e para realizar os seguintes comandos:

- Escolha uma posição no ambiente, que será considerada a posição inicial.
 - Gire 90° no sentido anti horário;
 - Mova 5 passos
 - Se tocar no obstáculo, então gire 90° para direita e mova 5 passos. Senão, volte à posição inicial.
- Escolha uma posição no ambiente;
 - Fique em pé;
 - Quando o professor colocar a mão sobre o ombro do aluno:
 - Repita 3x: Mude de traje (agache ou levante)

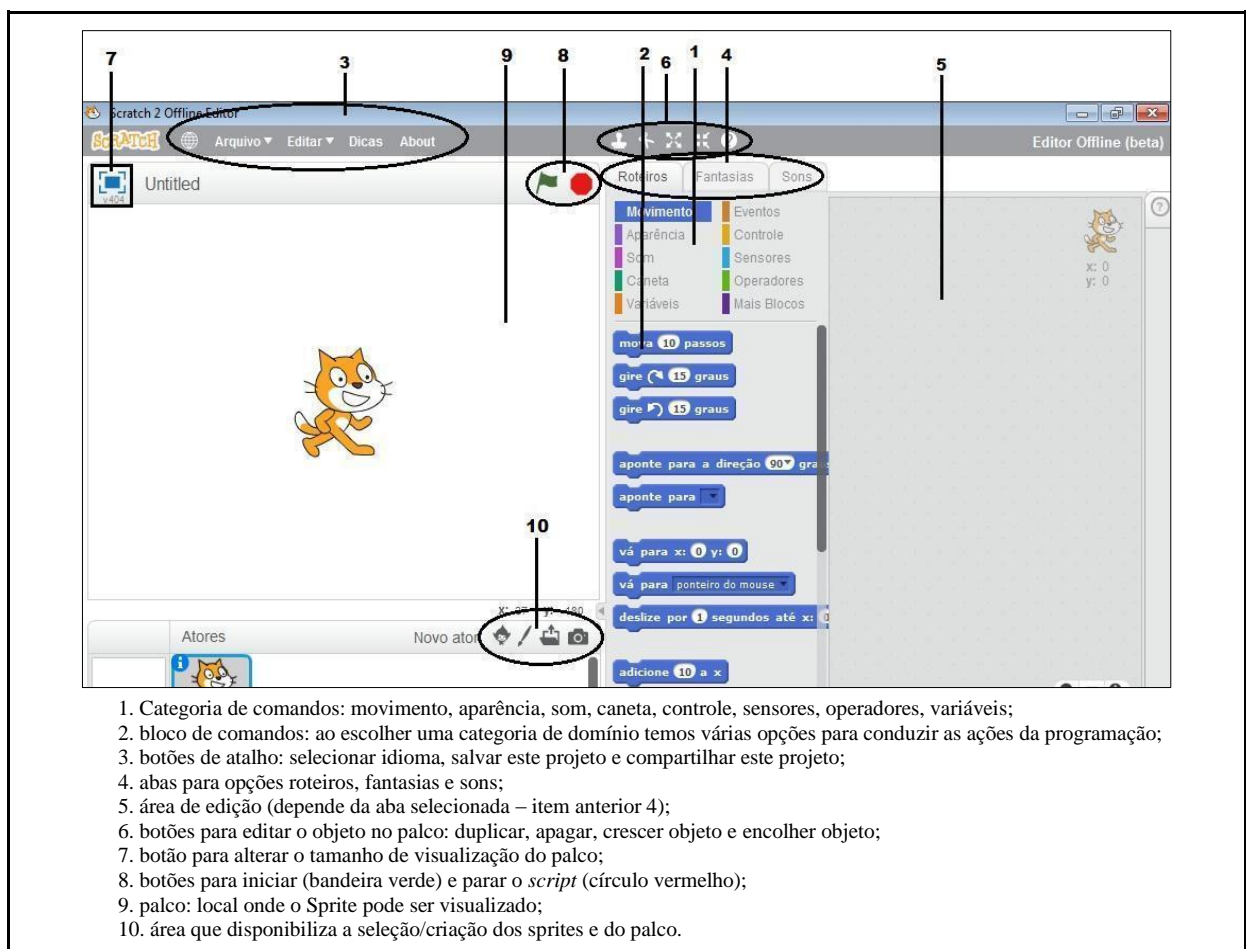
- Mova 5 passos
- Se tocar o objeto (classe ou cadeira), pare (terminar aqui)
- Quando ouvir um som:
 - Repita 6x: Mude para o traje 1 - abrir os braços e fechar as pernas;
 - Espere 2 segundos
 - Mude para o traje 2 - baixar os braços e abrir as pernas.
- Divida o grupo em duplas e solicite que eles criem uma encenação com os seguintes comandos:
 - Mova, gire e diga;
 - Mova, gire, se ... então;
 - Quando tocado, mude de traje e emita som.

c) Trocar uma lâmpada: em grupo escreva passo a passo os comandos necessários para trocar uma lâmpada. Escrever num cartaz. Todos os grupos devem apresentar para os colegas.

2 CONHECENDO O SOFTWARE SCRATCH

2.1 Tela Inicial

Vamos conhecer agora os nomes dos campos, abas, botões da Tela Inicial para facilitar o uso deste aplicativo. Na Figura aparece um objeto (o ator) com a imagem de um gato que pode ser movimentado na tela a partir da utilização de alguns comandos desta linguagem de programação.



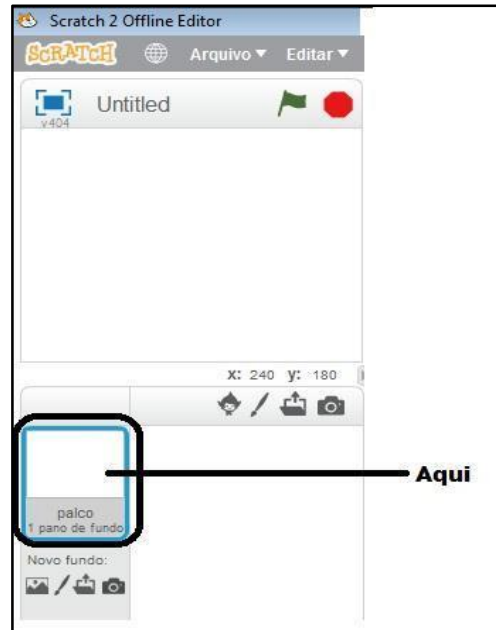
Tela Inicial do Scratch: áreas, botões, abas e elementos gráficos.

2.2 Palco

O Palco é o lugar onde pode ser visualizada a execução de eventos ou projetos; é o ambiente no qual os objetos estão inseridos. Pode ser, por exemplo, um fundo com uma paisagem, um ambiente que represente um local, um período como o dia, a noite ou qualquer contexto no qual a história, o jogo está inserido.

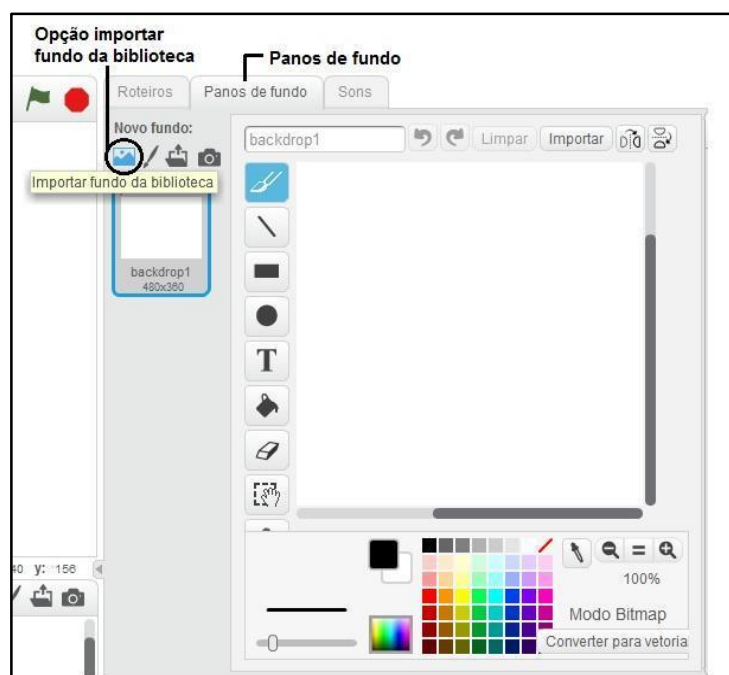
As dimensões do palco são 480 unidades de largura e 360 unidades de altura e está dividido entre as coordenadas X e Y. O centro do palco tem as coordenadas 0 em "X" e 0 em "Y".

Vamos, agora, aprender a buscar, nos arquivos, um palco já disponibilizado pelo Scratch.



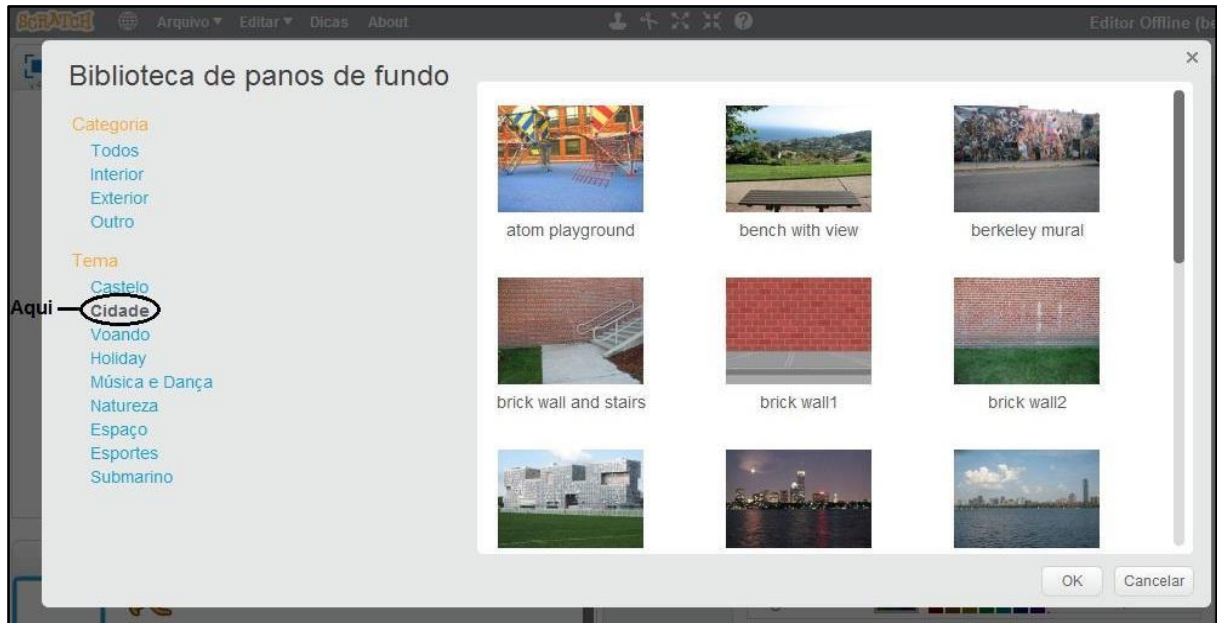
Ícone para acesso ao Palco.

Na tela inicial do Scratch, clique em “Panos de fundo” e em seguida no botão “Importar fundo da biblioteca”



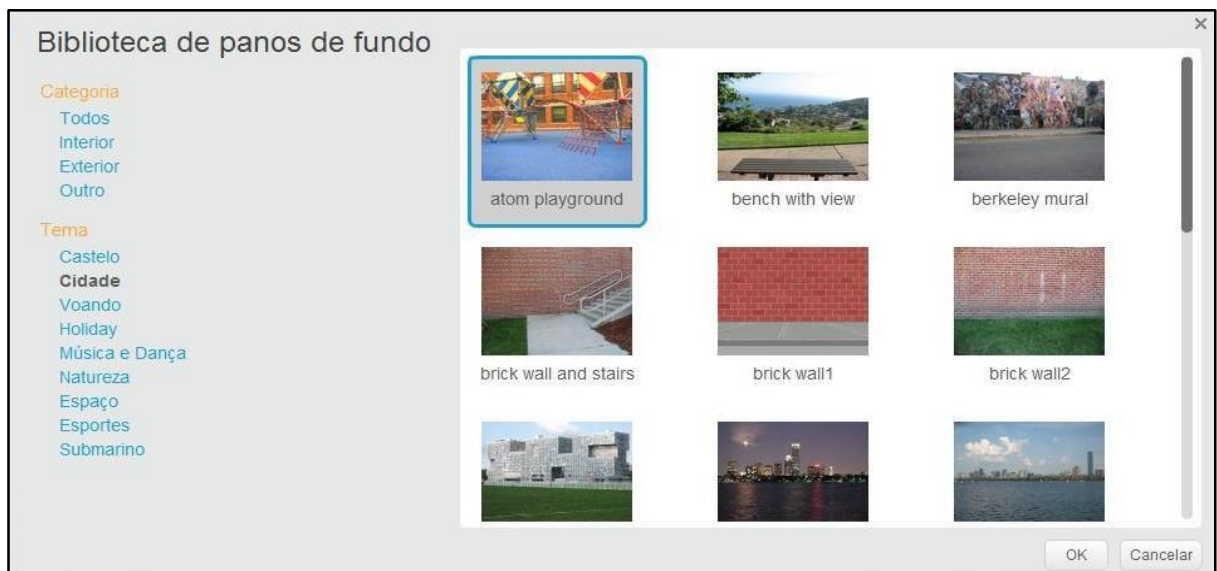
Ícones para importação de panos de fundo.

Em seguida, você terá acesso à janela mostrada na figura. Escolha o tema, por exemplo “Cidade”.



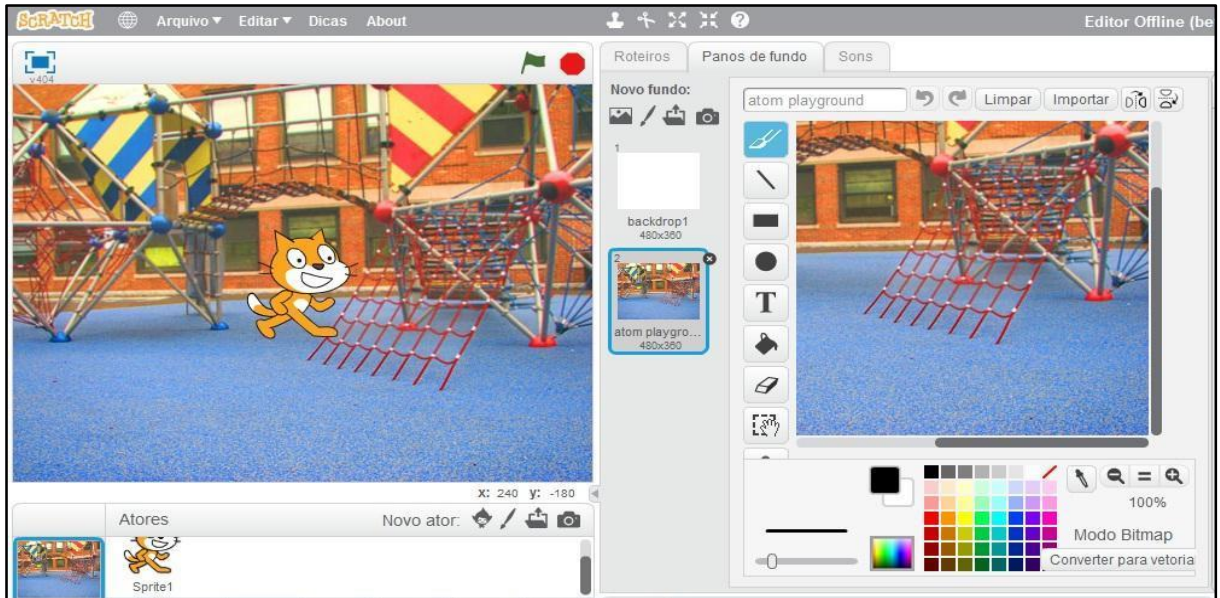
Janela com arquivos para importação.

Após entrar na pasta “Cidade” escolha a primeira opção, por exemplo.



Opções de planos de fundo, na pasta Cidade.

Assim, o seu palco será modificado automaticamente, conforme ilustrado pela figura.



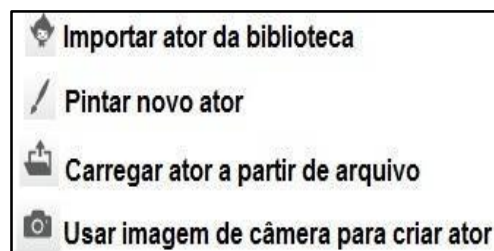
Novo pano de fundo inserido no palco.

2.3 Atores

O desenvolvimento de qualquer aplicação é feito usando objetos, denominados atores, que se constituem em imagens, desenhos, fotos ou qualquer outro objeto visual.

A aparência de um ator pode ser modificada pela sua apresentação em uma fantasia diferente. Qualquer imagem pode ser usada como uma fantasia; pode-se desenhar uma imagem no editor de Pintura, importar de uma lista ou baixá-la de um website. Você pode dar instruções a um ator, mandar que ele se mova, que toque música, reaja a outros atores, etc.

A tela inicial do Scratch, quando aberta, mostra como ator o Gatinho, conforme já ilustrado. Para criar novos atores, clica-se em um dos botões mostrados na figura.



Botões para criar ator.

A figura a seguir mostra as características do ator no momento que se está elaborando ou executando um projeto: nome, abscissa e ordenada, direção e o estilo de rotação.



Descrição das características do ator.

- Para dar um novo nome basta digitá-lo no retângulo que está escrito “ator 1”.
- A direção indica para onde ele se moverá quando executar um comando de deslocamento (0=para cima; 90=direita; 180=para baixo; -90=esquerda).
- A linha azul na miniatura também mostra a direção do ator. Arrastando essa linha com o mouse, a direção do ator, no Palco, será mudada; um clique duplo na miniatura devolverá a direção original (90).
- Pode-se exportar um ator clicando com o botão direito do mouse no ator do Palco ou da Lista de atores. Será guardado como um arquivo do tipo “ator” e poderá ser importado em outro projeto.

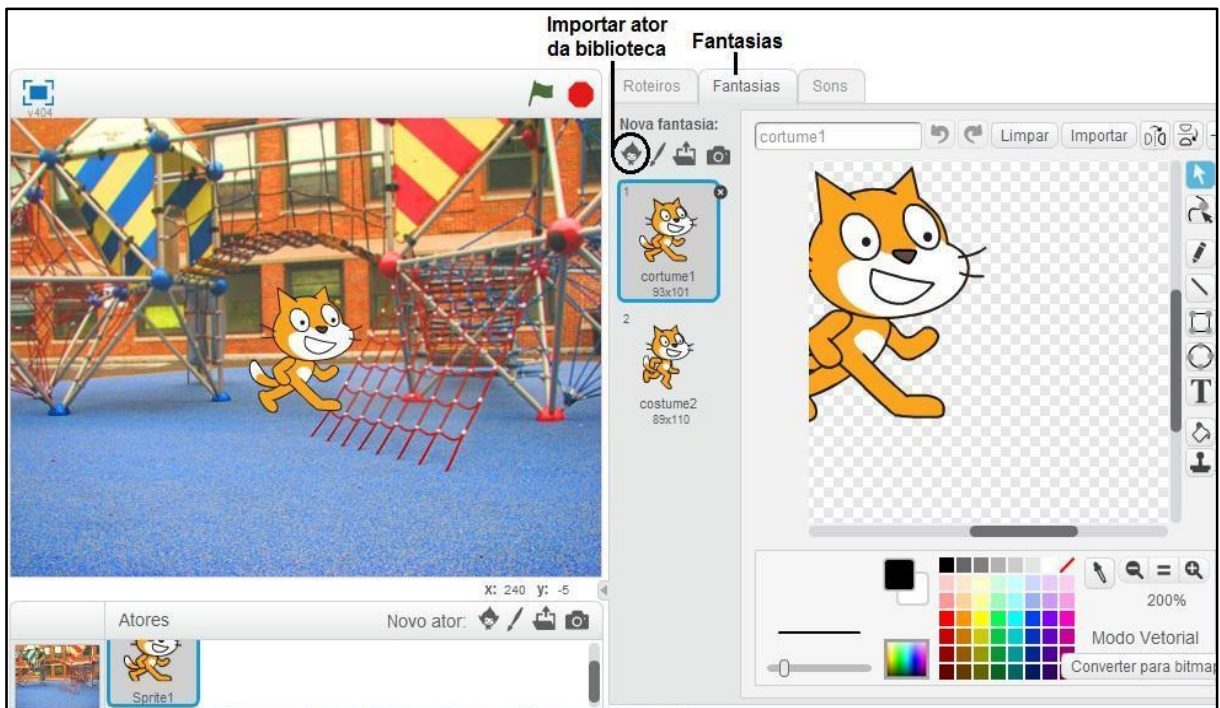
Vamos ilustrar como mudar a fantasia do Objeto 1:

- Clique no Objeto 1



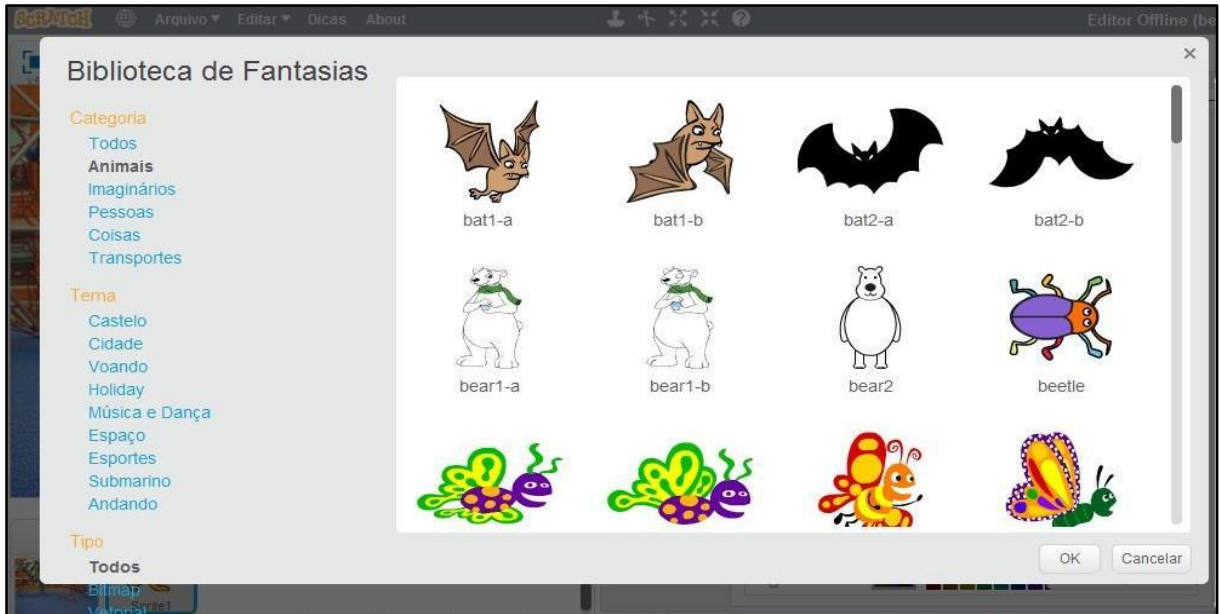
Clique no Objeto 1.

- Clique na aba “Fantasias” e em seguida clique no botão “Importar ator da biblioteca”:



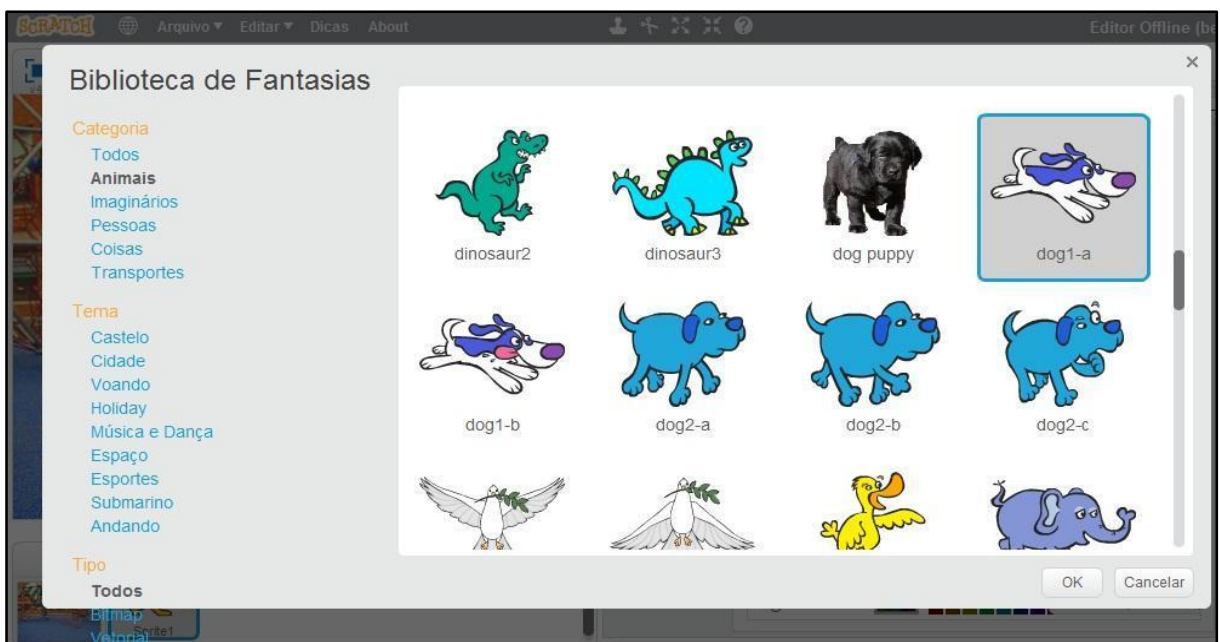
Importação de fantasias.

- Será aberta a janela abaixo, selecione o botão em destaque “Fantasias” e, em seguida, clique na pasta “Animais”, por exemplo,



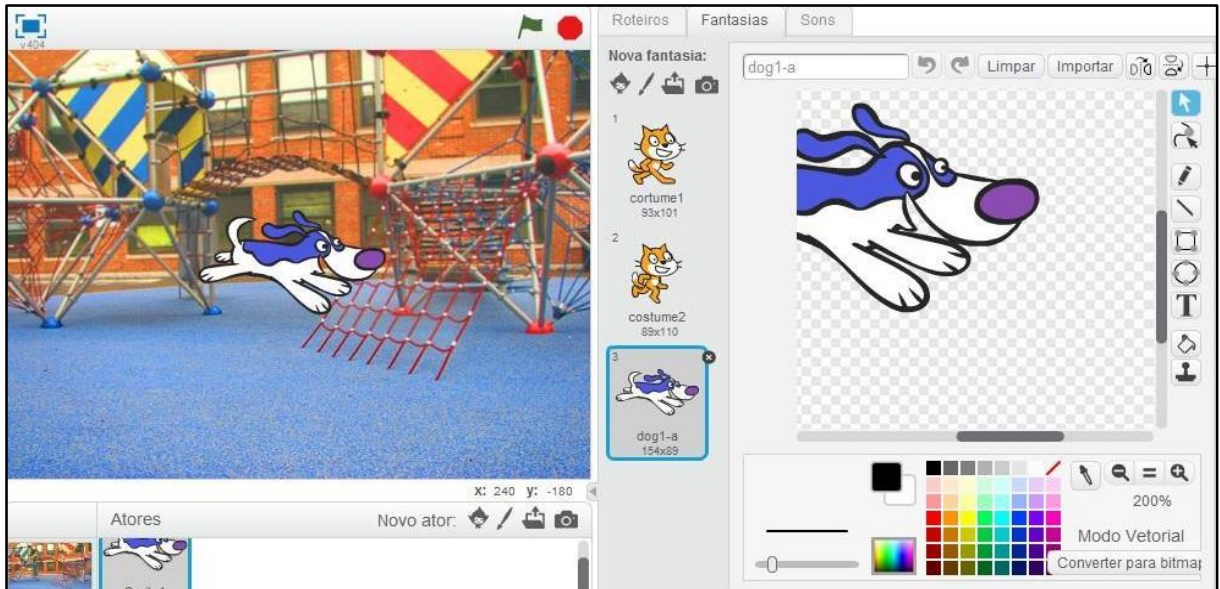
Arquivos contendo modelos de fantasias.

- Selecione o seguinte animal: o cachorro chamado “dog1-a”




Inserção do dog1-a.

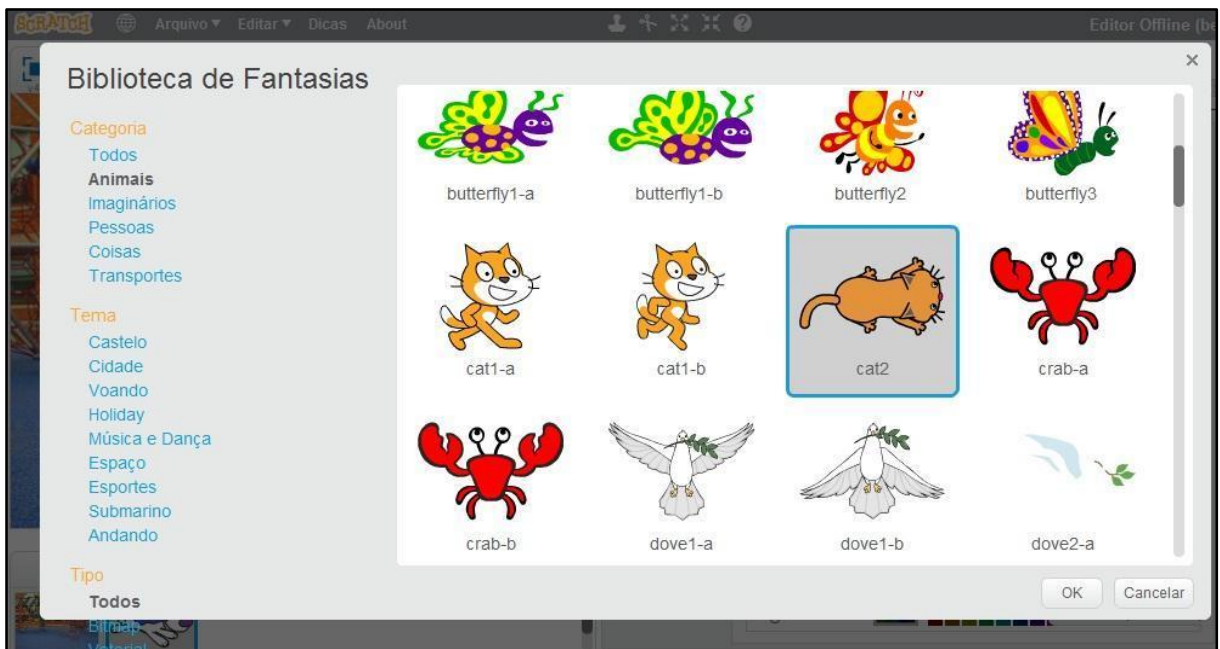
- Teremos, assim, a tela mostrada na figura:



Nova fantasia para o ator gatinho.

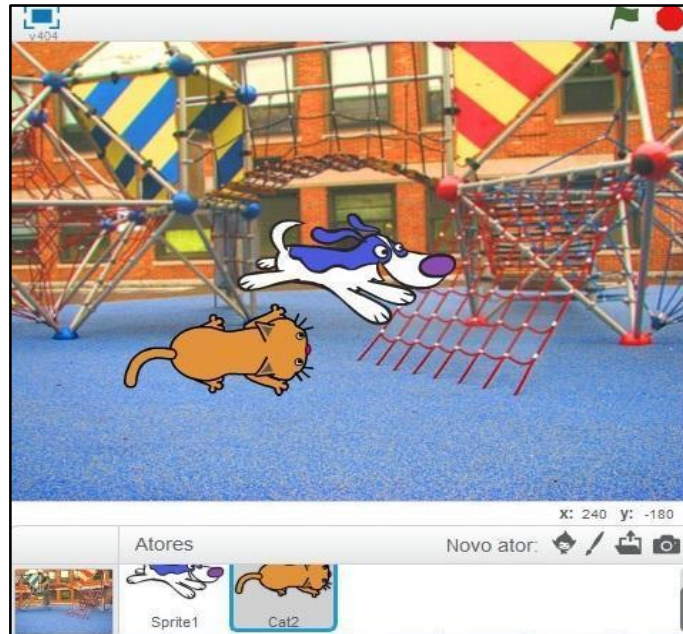
Aprendemos a trocar o gato pelo cachorro, agora, vamos inserir os dois no mesmo palco, para tanto, clique no ícone .

Teremos acesso aos objetos que estão em pastas. Clicamos na pasta “Animais” e vamos escolher outro gato:



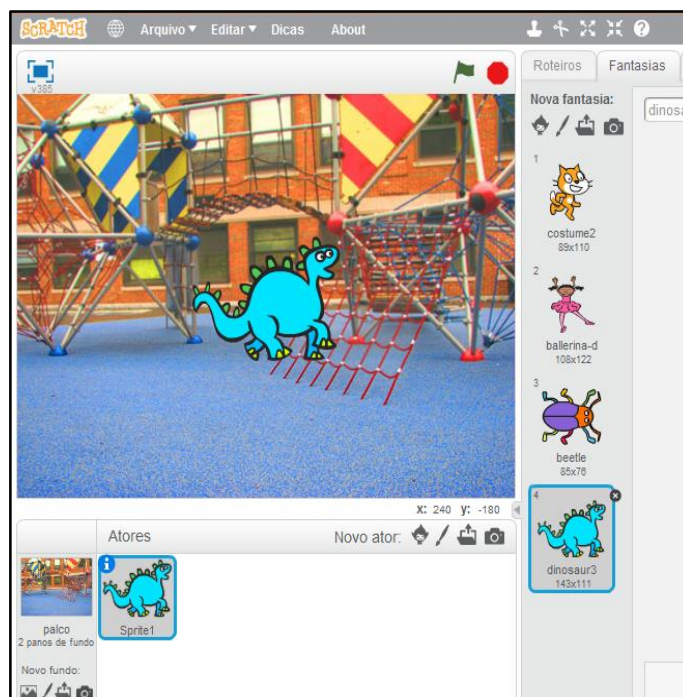
Menu de escolha de Atores.

- Agora temos dois atores no palco:



Dois atores no palco

A lista de atores contém miniaturas de todos os atores do projeto, mostrando o nome, quantas fantasias e quantos scripts (roteiros, fantasias e sons vinculados aos objetos) cada um possui. Para ver e editar o script de um ator clicar na miniatura na Lista de Atores ou dar um duplo-clique no ator na tela. O Palco pode mudar de aparência assumindo outros fundos de tela. Para ver e editar scripts, fundos de tela e sons associados ao Palco, clique no ícone do Palco (retângulo), à esquerda da Lista de Atores.



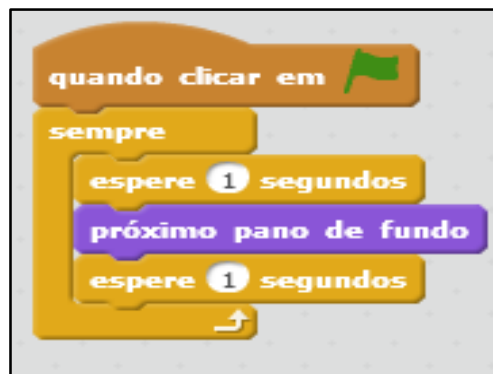
Lista dos atores usados no projeto.

Cada objeto utilizado na animação pode ter o seu comportamento programado a partir de um conjunto de blocos que se constituem em comandos que serão seguidos pelo personagem quando a animação for executada. Os comandos podem ser de movimento, por exemplo, que indicam que o personagem vai se movimentar x passos em uma certa direção, que vai até uma posição específica, etc. Outros comandos permitem modificar a aparência, a fantasia, cores e demais características dos objetos. Além destes existem estruturas de controle que permitem executar atividades de forma repetida, somente se uma condição for satisfeita, que algum evento ocorreu, etc.

2.3.1 Atividade 1

- Insira dois palcos e faça com que ocorra a troca entre eles.

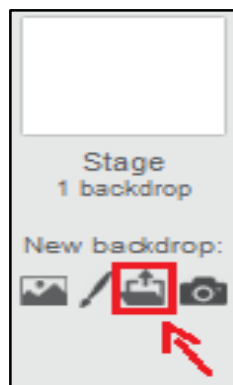
Salvar atividade em um arquivo.




Sugestão de comando para execução da atividade 1.

- Criar um palco (pintar um novo palco).
- Criar um palco importando a imagem da internet.



Obs.: Para importar a imagem da internet é preciso tê-la salva no computador e fazer a importação do computador clicando em:

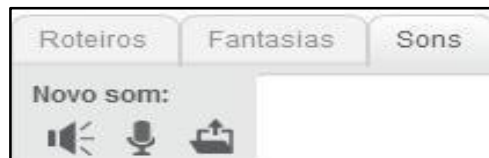


2.3.2 Atividade 2

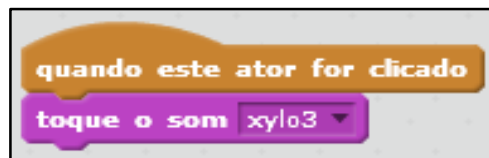
- Exclua o ator gatinho.
- Crie um Ator retangular usando o editor de pintura .
- Salvar atividade em um arquivo.

2.3.3 Atividade 3

- Criar um arquivo novo (vai aparecer o ator gatinho).
- Excluir o ator gatinho.
- Inserir um ator da biblioteca .
- Inserir um segundo ator da biblioteca .
- Clicar sobre um dos atores e inserir um som usando:



- Salvar atividade em um arquivo.



Sugestão de comando para execução da atividade 3.

2.3.4 Atividade 4

- Crie um novo arquivo.
- Exclua o ator gatinho.
- Insira um palco a sua escolha da pasta “natureza”.
- Escolha um ator com mais de uma fantasia e o insira no palco.
- Faça a mudança da fantasia do ator escolhido.



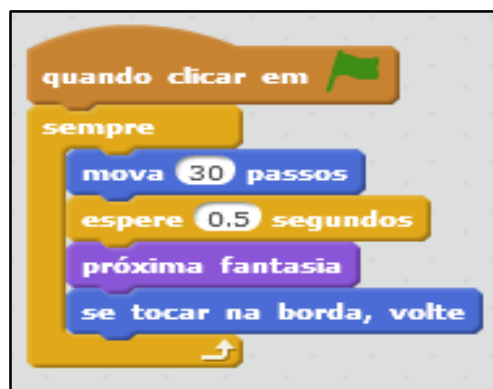
Sugestão de comando para execução da atividade 4.

2.3.5 Atividade 5

- Criar um arquivo novo.
- Salvar imagens da internet em um arquivo do computador.
- Importar a imagem salva no arquivo do computador como um novo ator, usando



- Usando o área de edição dos atores, altere a fantasia do ator;
- Inclua um palco de acordo com a figura escolhida.
- Finalize a atividade deixando o palco e os atores esteticamente apresentável e faça mudança de fantasia, utilizando as imagens salvas;



Sugestão de comando para execução da atividade 5.


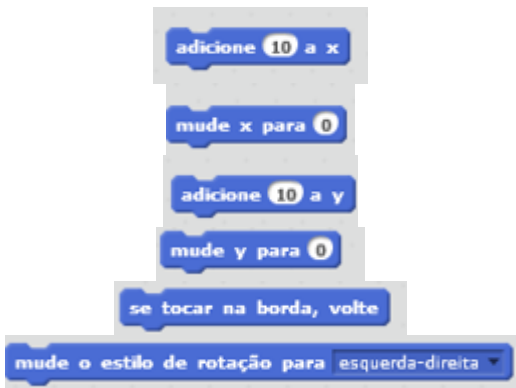
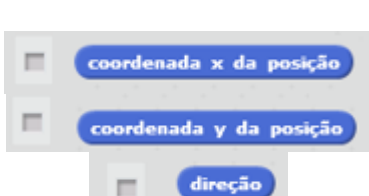
2.4 Roteiros

Neste item descrevemos os comandos necessários para programação. Para determinar o que cada ator vai executar, devem-se criar para cada um, sequências de roteiros, arrastando blocos gráficos e juntando-os em pilhas, chamadas script. Basta um duplo clique, em qualquer bloco, para fazer rodar um script. O Scratch executa esses scripts a partir do topo e depois, sucessivamente, bloco a bloco, os de baixo.

2.4.1 Roteiros de movimento


Abaixo são apresentados os comandos de movimento e a descrição básica de cada um.

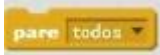
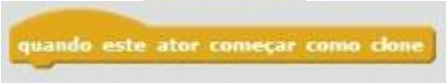


	Gira o ator 15 graus no sentido horário.
	Gira o ator 15 graus no sentido anti-horário.
	Aponta o ator para uma direção (90° direita, - 90° esquerda, 0° cima e 180° baixo)

	<p>Aponta para outro ator ou para o cursor.</p> <p>Move o ator para a posição indicada.</p> <p>Move o ator para outro ator ou para o cursor.</p> <p>Desliza o ator para a posição indicada, durante o tempo indicado.</p>
	<p>Adiciona a quantidade de passos no eixo x.</p> <p>Move o ator para a posição indicada no eixo x.</p> <p>Adiciona a quantidade de posições no eixo y.</p> <p>Move o ator para a posição indicada no eixo y.</p> <p>Faz com que o ator vire se tocar na borda.</p> <p>Muda o estilo de rotação para esquerda-direita, não girar ou em todas as posições.</p>
	<p>Se clicado, mostra a coordenada x na tela.</p> <p>Se clicado, mostra a coordenada y na tela.</p> <p>Se clicado, mostra a coordenada direção na tela.</p>

2.4.2 Roteiros de controle









Abaixo são apresentados os comandos de controle e a descrição básica de cada um.

	<p>Adiciona o tempo de espera indicado.</p> <p>Repete o número de vezes as instruções que estão dentro do bloco.</p> <p>Repete sempre as instruções que estão dentro do bloco.</p> <p>Se a condição for verdadeira, então execute as instruções que estão dentro do bloco.</p> <p>Se a condição for verdadeira, então execute as instruções que estão dentro do bloco se, caso contrário, execute as instruções que estão no bloco senão.</p> <p>Espere até que a condição seja verdadeira para executar o bloco seguinte.</p> <p>Repete as instruções do bloco até que a condição esteja satisfeita.</p>
---	---

	Envia uma mensagem para todos os atores.
	Diz ao clone o que fazer, quando o mesmo for clicado.
	Cria um clone do ator especificado.
	Apaga o clone atual.



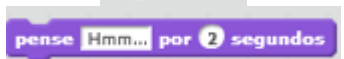

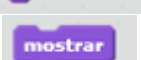
2.4.3 Roteiros relativos à eventos




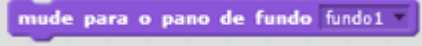



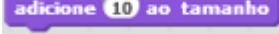






Abaixo são apresentados os comandos de eventos e a descrição básica de cada um.

	Quando bandeira verde for clicada, inicia as interações que estão no bloco.
	Quando tecla selecionada for pressionada, inicia as interações que estão no bloco.
	Quando clicar no ator, inicia as interações que estão no bloco.
	Executa um script quando o cenário muda para um determinado fundo.
	Executa um script quando o atributo selecionado (loudness, timer, movimento de vídeo) é maior que um valor especificado.
	Executa um script quando recebe mensagem de transmissão especificada.
	Envia uma mensagem para todos os Atores.
	Envia uma mensagem para todos os Atores e aguarda.

3.4.4 Roteiros relativos à aparência

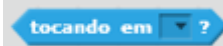



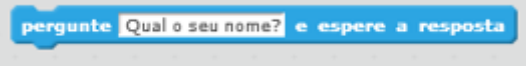




Abaixo são apresentados os comandos de aparência e a descrição básica de cada um.








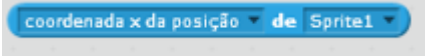



	Mostra uma mensagem em um balão do tipo fala durante o tempo indicado.
	Mostra uma mensagem em um balão do tipo fala.
	Mostra uma mensagem em um balão do tipo pensamento durante o tempo indicado.
	Mostra uma mensagem em um balão do tipo pensamento.
	Mostra o Ator.

	Esconde o Ator.
	Muda a fantasia do ator para a fantasia selecionada.
	Muda a fantasia do ator para a próxima fantasia da lista.
	Muda o pano de fundo para o pano de fundo selecionado.
	Adiciona um efeito.
	Define um efeito gráfico.
	Apaga os efeitos gráficos
	Muda o tamanho do Ator.
	Muda o tamanho do Ator.
	Move o Ator para a frente dos outros Atores.
	Move o Ator para trás o número de camadas indicadas.
<input type="checkbox"/> 	Se marcado, mostra o número de fantasias do Ator na tela.
<input type="checkbox"/> 	Se marcado, mostra o nome do pano de fundo na tela.
<input type="checkbox"/> 	Se marcado, mostra o tamanho na tela.

3.4.5 Roteiros que se constituem em sensores

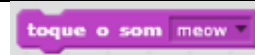

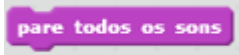


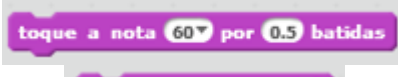
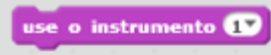




Abaixo são apresentados os comandos relativos a sensores e a descrição básica de cada um.

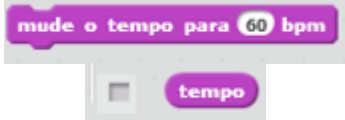
	Se o Ator tocar no objeto selecionado, retornará verdadeiro.
	Se o Ator tocar na cor selecionada, retornará verdadeiro.
	Se a cor selecionada estiver tocando na outra cor, retornará verdadeiro.
	Retorna a distância do Ator até o objeto selecionado.
	Aguarda até receber uma resposta.
<input type="checkbox"/> 	Recebe uma resposta e mostra na tela, se usada em conjunto com a instrução “diga”.
	Se tecla selecionada estiver pressionada, retornará verdadeiro.
	Se o mouse estiver pressionado, retornará verdadeiro.
	Se o mouse se movimentar no eixo x, retornará verdadeiro.

	Se o mouse se movimentar no eixo y, retornará verdadeiro.
	Relata o volume (de 1 a 100) dos sons detectados pelo microfone do computador.
	Sentidos quanto ao movimento ou direção e é atualmente a imagem de vídeo.
	Liga ou desliga o vídeo.
	Define a transparência do vídeo
	Relata o valor do temporizador em segundos.
	Zera o temporizador.
	Retorna o valor referente a opção selecionada.
	Informa a hora atual.
	Informa o número de dias desde 2000.
	Mostra o nome de usuário do projeto.

2.4.6 Roteiros para manipulação de som

















Abaixo são apresentados os comandos de manipulação de som e a descrição básica de cada um.

	Reproduz o som selecionado e continua as interações.
	Reproduz o som selecionado e espera até que este termine de ser reproduzido para seguir as interações.
	Interrompe a reprodução de todos os sons.
	Reproduz o som selecionado durante o intervalo de tempo indicado.
	Para a reprodução durante o intervalo de tempo indicado.
	Reproduz a nota selecionada durante o intervalo de tempo indicado.
	Usa o instrumento indicado.
	Adiciona o valor indicado ao volume do programa.
	Muda o volume do programa. Também pode ser usado juntamente com um operador.
	Se marcado, mostra o volume de programa. Também pode ser usado juntamente com um operador.
	Adiciona o valor indicado ao número de batidas por minuto do som.

	Muda o número de batidas por minuto para o valor indicado. Se marcado, mostra o tempo na tela.
---	---



2.4.7 Roteiros que implementam operadores


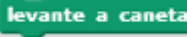
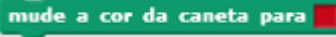





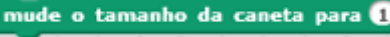
Abaixo são apresentados os comandos de implementação de operadores e a descrição básica de cada um.

	Adição.
	Subtração.
	Multipliação.
	Divisão.
	Escolha um número inteiro dentro do intervalo especificado.
	Verifica se o 1º valor é menor que o 2º.
	Verifica se o 1º valor é igual ao 2º.
	Verifica se o 1º valor é maior que o 2º.
	Retorna verdadeiro se as duas condições estiverem corretas.
	Retorna verdadeiro se ao menos uma das condições estiverem corretas.
	Altera a condição, se for verdadeira se tornará falsa; se for falsa, se tornará verdadeira.
	Conecta duas frases.
	Retorna a letra na posição escolhida.
	Retorna o resto da divisão de um número por outro.
	Arredonda o valor desejado.
	Retorna o resultado da função matemática escolhida (Raiz quadrada, seno, cosseno).

2.4.8 Roteiros para uso da caneta

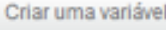
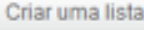
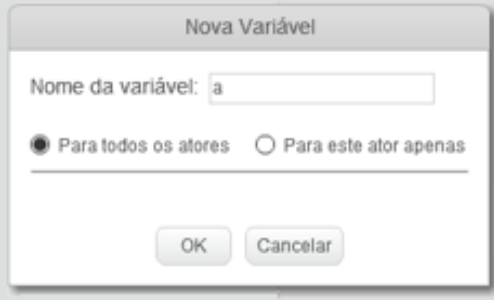
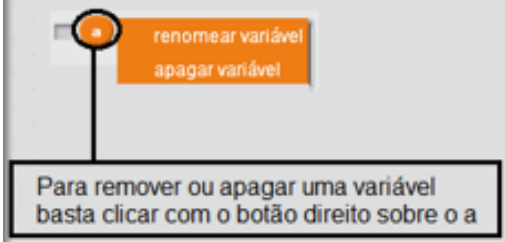
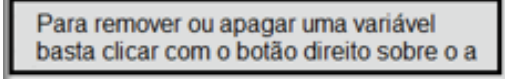




Abaixo são apresentados os comandos para uso da caneta e a descrição básica de cada um.

	Limpa todas as marcas deixadas pela caneta na tela.
	Carimba na tela o próprio Ator.

	Usa a caneta.
	Interrompe a marcação da caneta.
	Abre uma caixa de cores para escolher um nova cor para a caneta.
	Adiciona o valor indicado ao número da cor da caneta.
	Muda a cor da caneta para a cor correspondente ao valor escolhido.
	Adiciona o valor indicado à intensidade da caneta.
	Muda a intensidade da caneta para o valor indicado.
	Adiciona o valor indicado à espessura do risco da caneta.
	Muda a espessura do risco da caneta para o valor indicado.

2.4.9 Roteiros para utilização de variáveis

Abaixo são apresentados os comandos para utilização de variáveis e a descrição básica de cada um.

	Cria uma nova variável.
	Cria uma lista.
	
	
	
	Se marcado, mostra a variável na tela.
	Muda o valor da variável para o valor indicado.
	Adiciona o valor indicado ao valor da variável.
	Mostra a variável na tela.
	Esconde a variável.

Para remover ou apagar uma variável basta clicar com o botão direito sobre o a







Inserir o valor indicado na lista.

Apaga o item escolhido da lista.

Inserir o valor na posição indicada.

Substituir um valor na lista.

Retorna item escolhido da lista.

Retorna tamanho da lista.

Verifica se o item escolhido está contido na lista.

Mostra a lista b.

Esconde a lista b.

3 TRABALHANDO NO SCRATCH

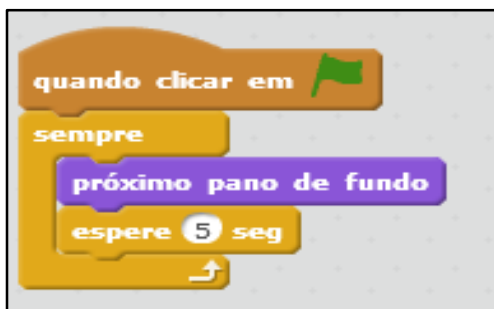
- Fazer a introdução dos blocos, explicando pontos importantes;
- Quando necessário introduzir um texto explicando, por exemplo, o que é variável, conceito listas, clone, cronômetro...
- Inserir atividades que envolva vídeo.

Este tópico contém atividades e exercícios para serem desenvolvidos no software Scratch, que envolvem os conceitos estudados nos blocos anteriores. Os exercícios apresentados aqui foram elaborados e executados pelos professores e acadêmicos que participam do Projeto Escola de Hackers e têm como objetivo desenvolver a lógica da programação, bem como a compreensão de programar no Scratch.

- **Proposta metodológica:** Apresentar de modo geral o software onde se encontram os roteiros utilizados para a programação e iniciar a realização das atividades. Caso houver dúvida quanto o significado de algum comando ou bloco de comandos, volte ao bloco 1- conhecendo o Scratch.

3.1 Atividade 1 - ATOR E PALCO

- Explicar novamente, caso seja necessário, sobre os comandos básicos de ator e palco
- Inserir dois atores diferentes, que possuam ao menos duas fantasias cada (por exemplo, o gatinho e o morcego), e dois panos de fundo diferentes;
- Fazer com que os panos de fundo se alternem a cada 5 segundos;
- Fazer com que os atores mudem de fantasia a cada 0.2 segundos;

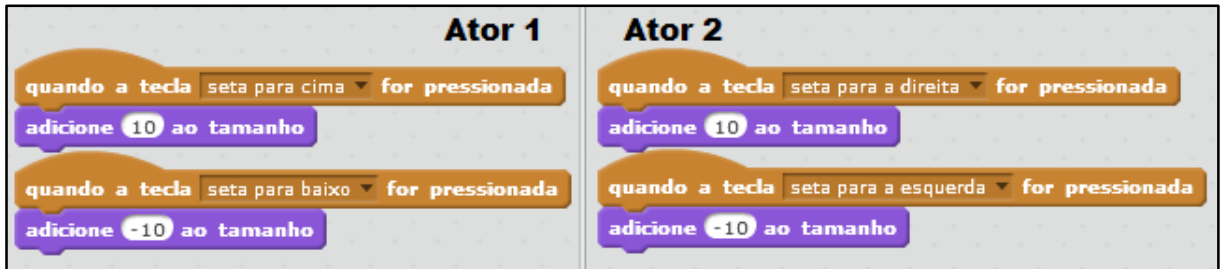


Sugestão de comando do **palco** para execução da atividade 1.



Sugestão de comando dos **atores** para execução da atividade 1.

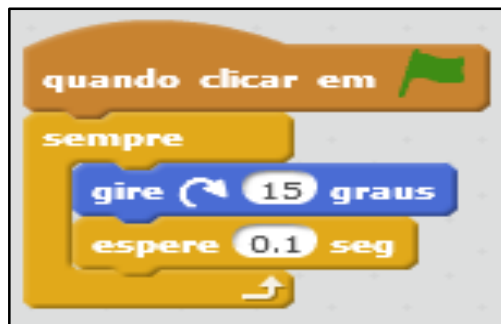
- Extras
 - O ator 1 deverá aumentar seu tamanho quando a seta para cima for pressionada e diminuí-lo quando seta para baixo for pressionada;
 - O ator 2 deverá aumentar seu tamanho quando a seta para direita for pressionada e diminuí-lo quando seta para esquerda for pressionada;



Sugestão de comando do palco para execução da atividade 1 extra.


3.2 Atividade 2 - CENTRO DE ROTAÇÃO

- Aumente algumas vezes o tamanho do ator gatinho;
- Crie um roteiro para que sempre, a cada 0.1 segundo, o ator gire 15 graus para qualquer um dos lados;



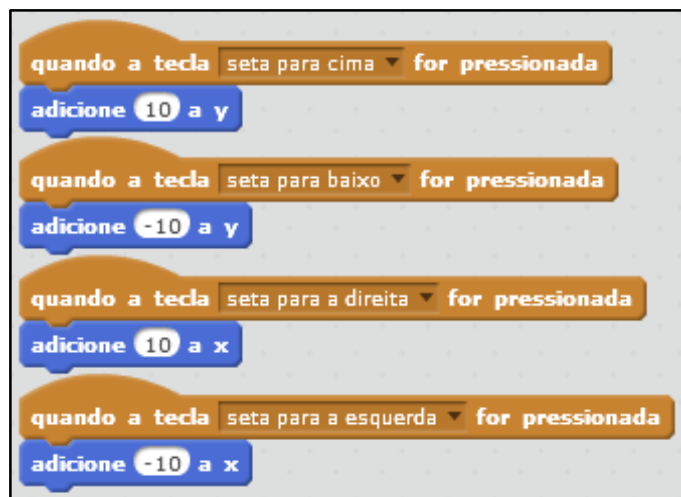
Sugestão de comando para execução da atividade 2.

- Agora clique na bandeira para o roteiro ser executado, e enquanto o ator gira, altere seu centro de rotação, testando:
 - Na ponta do pé;
 - Na orelha;
 - Na ponta da cauda;
 - No peito.
- Dica para mudar o centro de rotação do objeto:
 - Selecione o ator
 - Vá para a aba “Fantasias”

- Clique em , que se encontra no canto superior direito.
- Desta forma, pode-se localizar o centro do ator, bem como mudar seu ponto de rotação, basta arrastar o centro da cruz até o ponto desejado.

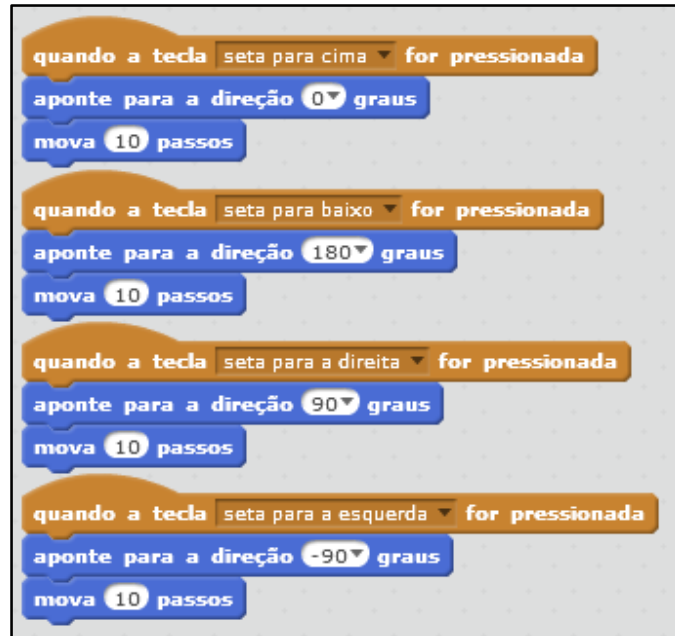
3.3 Atividade 3 - MOVIMENTO DO ATOR USANDO AS SETAS

- Inserir um ator;
- Insira o pano de fundo “xy-grid” para ajudar os alunos na orientação das coordenadas;
- Fazer o ator se movimentar quando as setas de direção (direita, esquerda, para cima, para baixo) forem clicadas;
- Sugira que inicialmente seja feito utilizando as coordenadas X e Y:



Sugestão de comando para execução da atividade 3.

- Extra
- Extra: Feito isso, peça aos alunos que adicionem outro ator (ex.: morcego), e façam com que ele se mova virando o ator para os lados e andando em passos:



Sugestão de comando alternativo para execução da atividade 3 extra.

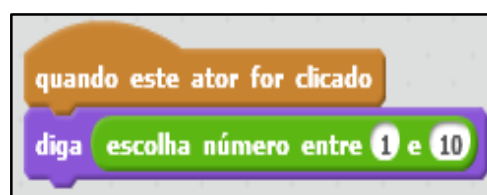
- Você pode sugerir que os atores mudem de fantasia cada vez que se mexerem!
- Observem que os atores estarão se movendo de cabeça para baixo, para finalizar instrua os alunos para que fixem a rotação do ator na horizontal:



Sugestão de comando para execução da atividade 3 extra.

3.4 Atividade 4 - LOTERIA

- Exclua o ator gatinho;
- Insira um ator botão;
- Quando o botão for clicado ele deverá sortear um número aleatório e mostrá-lo na tela;
- Inicie usando valores pequenos (ex.: 1 até 10) e então vá usando valores mais amplos (ex.: 1 até 1000, -1000 até 1000, etc).



Sugestão de comando para execução da atividade 4.

3.5 Atividade 5 - DESENHAR INICIAL

- Insira um ator lápis;
- Defina o centro de rotação do ator para a ponta do lápis e seu estilo de rotação para que ele não gire;
- Mude o tamanho da caneta para 5;
- Faça com que o ator desenhe a primeira letra de seu nome, usando comandos de “gire” e “mova”;
- Quando terminar, mova o lápis para a direita da letra e carimbe o lápis.

- Dica de como utilizar a caneta:

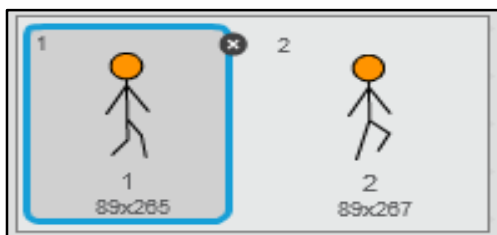
- Apague tudo: limpa tudo que foi feito com a caneta;
- Carimbe: cria um “carimbo” do ator no lugar onde ele se encontra;
- Use a caneta: “abaixa a caneta” do palco, para começar a desenhar quando o ator se mover;
- Levante a caneta: “levanta a caneta” do palco, para parar de desenhar quando o ator se move;
- As demais opções se referem à mudança de cor e tamanho do traço.



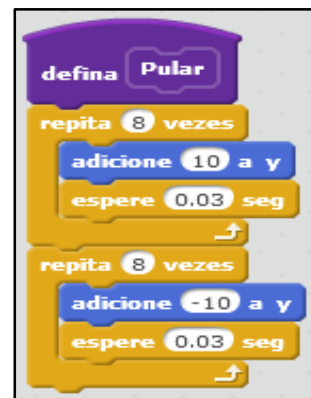
Sugestão de comando para execução da atividade 5, desenhando a letra “A”.

3.6 Atividade 6 - MAIS BLOCOS

- ❑ Desenhe um ator de palitinhos, com duas fantasias, como se estivesse andando.
- ❑ Agora iremos criar duas funções, uma para o ator andar e outra para ele pular: vá para a aba “Mais Blocos” e selecione a opção de criar um bloco.
- ❑ Primeiro, crie um bloco chamado “Pular” e clique em “ok”;
- ❑ Depois crie um bloco chamado “Andar”, clique em “opções” e adicione um campo de entrada de texto chamado “direção” e clique em “ok”;
- ❑ Quando clicar na bandeira verde, o ator deverá ir para a posição $X = 0$, $Y = 0$; e sempre:
 - Testar se as setas direcionais do teclado estão pressionadas (um teste para cada tecla); Se uma estiver pressionada, o ator deverá chamar a função “Andar”, passando no bloco o parâmetro da direção que deve se mover.
 - Testar se a tecla espaço está pressionada; Se estiver, o ator deverá pular.
- ❑ Para o bloco “Pular”, o ator deverá se mover para cima e para baixo usando apenas as coordenadas X e Y.
- ❑ Para o bloco “Andar” o ator deverá, baseado no parâmetro recebido, virar para o lado adequado, trocar de fantasia e se mover alguns passos.



Sugestão de modelos de fantasias para o ator.



Sugestão de comando para o bloco Pular.



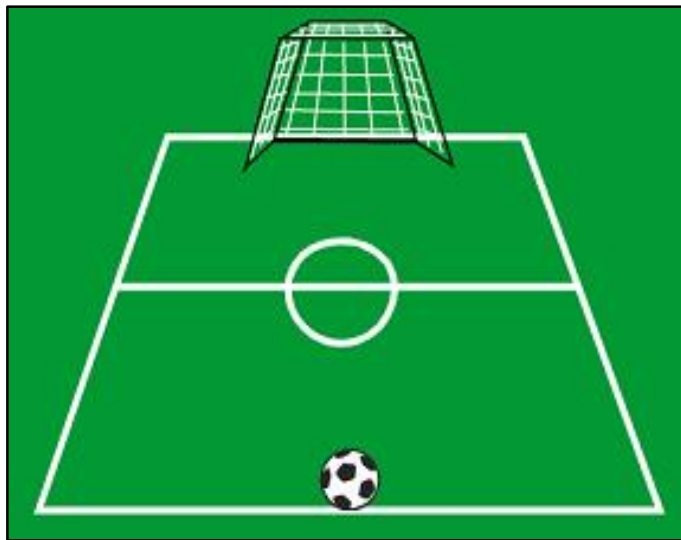
Sugestão de comando para execução da atividade 6.

4 USO DE CONDICIONAL - ROTEIRO CONTROLE

4.1 Atividade 7 - GOLEIRA

Agora vamos usar o teste completo: se, então... senão. O desafio é “chutar” uma bola na direção do gol e se acertar nele, dizer “Gol!”.

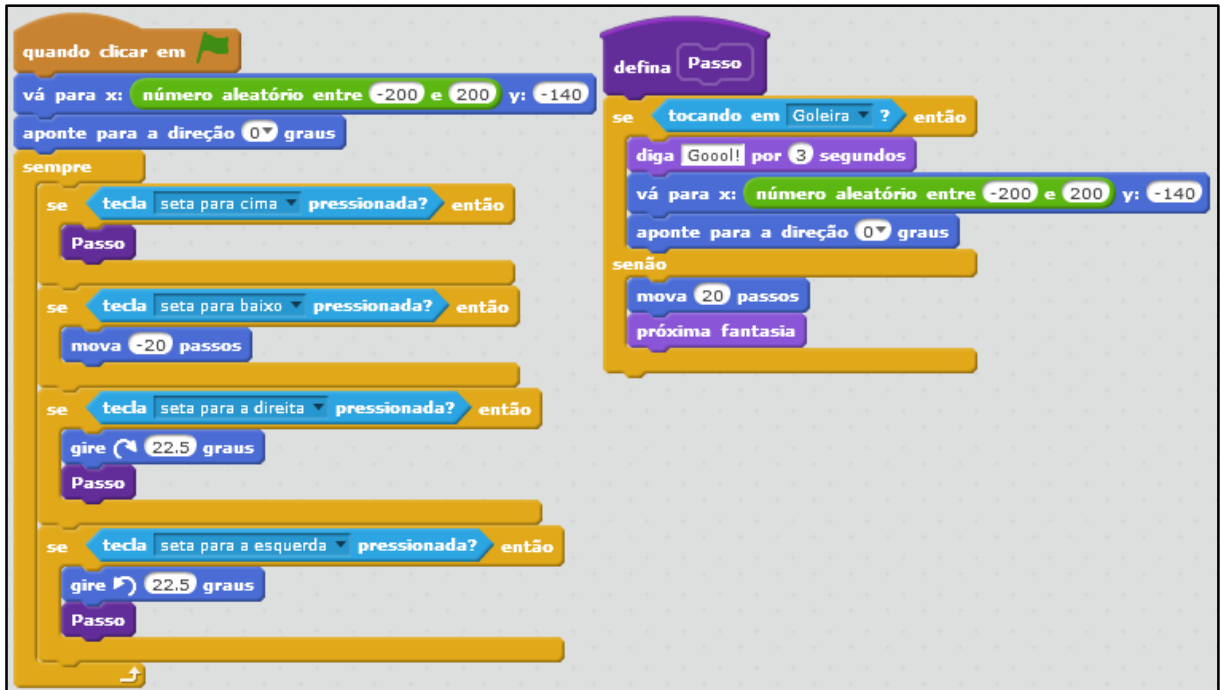
- ❑ Inicialmente pinte um pano de fundo de verde, será o gramado. Então desenhe um ATOR goleira e selecione um ator bola da biblioteca. Renomeie estes atores para “goleira” e “bola”, respectivamente;



Como os atores Bola e Goleira devem ser desenhados e posicionados.

- ❑ Faça que quando a bandeira verde for clicada a bola vá para baixo da tela, em uma posição X aleatória (entre -200 e 200), apontando para cima; e a goleira vá para sua posição, no topo da tela (exemplo: $x = 0$, $y = 90$)
- ❑ A partir de agora, sempre que alguma das setas do teclado forem pressionadas, a bola deverá se mover para o lado correspondente:
 - Se a seta for para baixo a bola deverá mover-se para trás (você pode usar -20 passos), sem alterar sua direção;
 - Se a seta for para a direita ou esquerda, a bola deverá girar 22.5 graus para o lado adequado, e andar;
 - Se a seta for para cima, o ator deverá apenas andar;
- Para andar, use o “Mais Blocos”, crie um bloco que testa se o ator está tocando na goleira:

- Sim - O ator deverá dizer “Goooool!” por 2 segundos e então voltar à posição inicial (ao sul da tela, com a coordenada X aleatória e apontando para cima);
- Não - O ator deverá mover-se 20 passos e trocar para a próxima fantasia;



Sugestão de comando para execução da atividade 7.

- Extra
 - Você pode sugerir aos alunos que façam com que o palco pisque algumas vezes quando for feito um gol.
 - Para o palco piscar, faça com que quando for feito o gol, a bola envie para todos uma mensagem chamada “Gol”
 - Quando o palco receber esta mensagem, ele deverá aumentar seu efeito de cor em 15, por 10 vezes
 - Por último, apague os efeitos gráficos para as cores voltarem ao normal



Sugestão de comando para execução da atividade 7 extra.

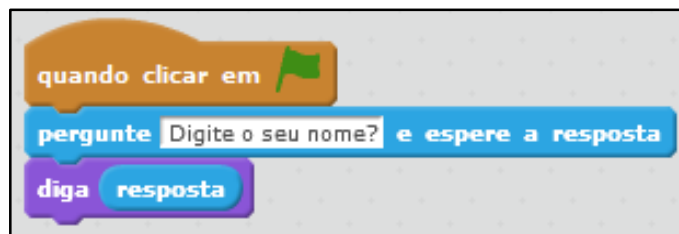
- **Noção de Variável**

No mundo computacional é possível armazenar dados numéricos ou sequências de caracteres (por exemplo palavras ou textos) em “unidades” chamadas de Variáveis. Podemos compará-las, de um modo simples, aos potes do mundo real, onde podemos armazenar alguma coisa, como um texto ou um número. Usando uma variável você pode armazenar e consultar valores definidos anteriormente, também realizar operações e cálculos sobre esta variável.

O Scratch suporta variáveis globais (que todos os atores terão acesso) e variáveis locais (específicas para um ator). Além disso, ele tem em sensores uma variável padrão denominada de "resposta". Nela é armazenado o último conteúdo respondido ao bloco “pergunte [...]”.

4.2 Atividade 8 - VARIÁVEL E RESPOSTA

- O ator deverá perguntar ao usuário qual é seu nome. Use os blocos de pergunta e resposta que se encontram na aba “sensores”.
- Faça com que o ator diga o que foi respondido.



Sugestão de comando para execução da atividade 8.

- Agora faça com que o ator pergunte o nome e depois o bairro em que a pessoa mora, e depois diga as duas respostas.
- Para isto, na aba de variáveis, crie uma variável para armazenar cada valor e após perguntar armazene a resposta na variável correspondente.
- O ator deverá dizer cada resposta durante 2 segundos.

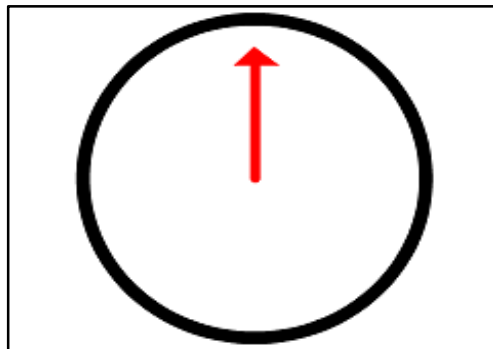


Sugestão de comando para execução da atividade 8.

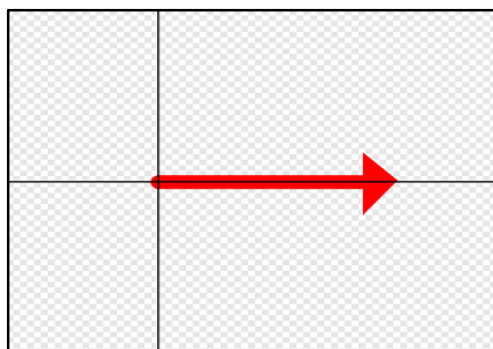
Dica: A variável “resposta” armazena apenas a ÚLTIMA resposta do bloco “pergunte”.

4.3 Atividade 9 - VARIÁVEL DE CONTAGEM

- No palco, desenhe um círculo grande no centro;
- Exclua o ator Gatinho e desenhe um ator em forma de seta colorida. O resultado deve estar parecido com isto:

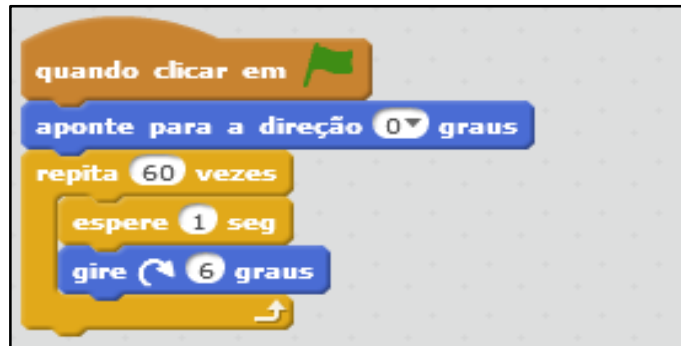


- Na aba “Fantasias”, altere o centro de rotação da seta para o lado oposto da ponta, como no exemplo:



Localização do centro de rotação do ator.

- ❑ Faça com que quando a bandeira verde for clicada, o ator aponte para cima e repita 60 vezes esta sequência de comandos:
 - Espere 1 segundo;
 - Gire 6 graus no sentido horário;



Sugestão de comando para execução da atividade 9.

- Extra
 - ❑ Você pode fazer o relógio mostrar os segundos em que se encontra, para isto, dê um pequeno conceito de variáveis (não entre a fundo no assunto pois variáveis serão estudadas nas atividades futuras) e instrua os alunos a construir a seguinte sequência de comandos:

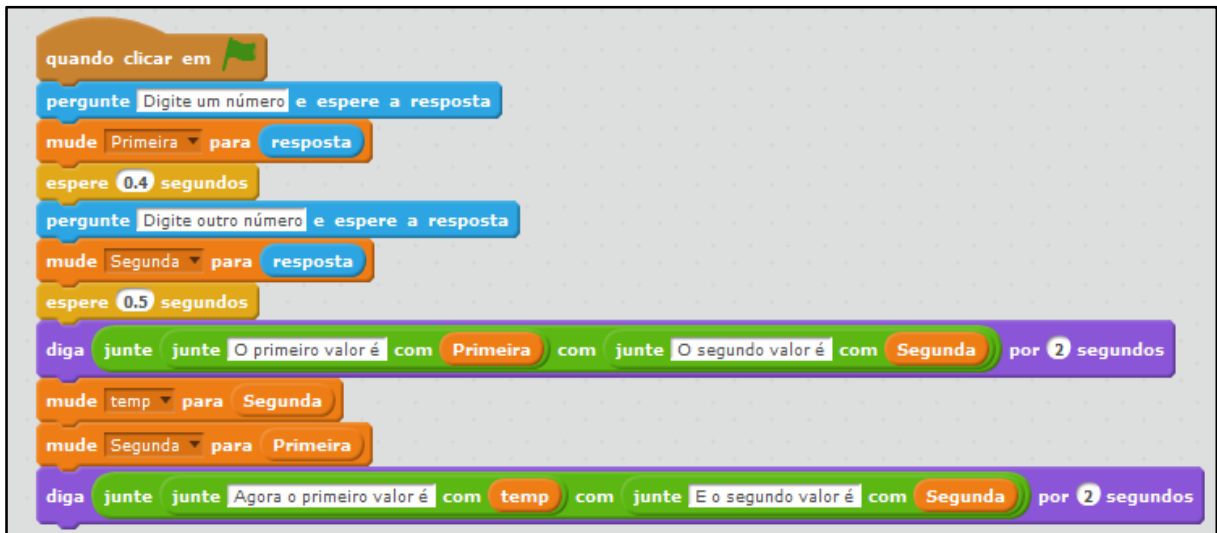


Sugestão de comando para execução da atividade 9 extra.

4.4 Atividade 10 - JUNTE

- ❑ Crie duas variáveis, chamadas PRIMEIRA e SEGUNDA;

- Faça o gatinho perguntar ao usuário que informe dois valores e atribua-os um em cada variável;
- A seguir, troque os valores, de forma que PRIMEIRA contenha o valor de SEGUNDA e vice-versa.
- O gatinho deve repetir os novos valores.

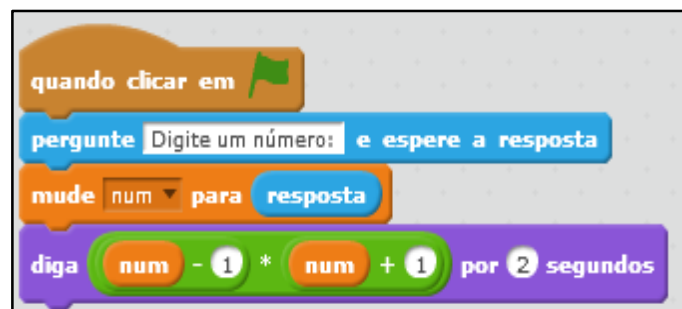


Sugestão de comando para execução da atividade 10.

- **Funções e Operações Matemáticas**

4.5 Atividade 11 - PRODUTO

- Faça o ator perguntar ao usuário que digite um número como entrada, e grave a resposta em uma variável;
- Calcular o produto de seu antecessor com o seu sucessor, $(n-1)*(n+1)$;
- Faça o gatinho (ator) dizer o resultado.



Sugestão de comando para execução da atividade 11.

4.6 Atividade 12 - MÉDIA ARITMÉTICA

- Crie 3 variáveis, chamadas “valor1”, “valor2” e “valor3”.
 - Faça o ator perguntar o valor para cada uma delas, não esqueça de armazenar as respostas nas respectivas variáveis;
 - Faça o ator dizer o o resultado da a média aritmética dos valores.
- Média aritmética: é a divisão da soma dos valores observados pela quantidade de valores envolvidos. Neste caso: $(\text{valor1} + \text{valor2} + \text{valor3}) / 3$.



Sugestão de comando para execução da atividade 12.

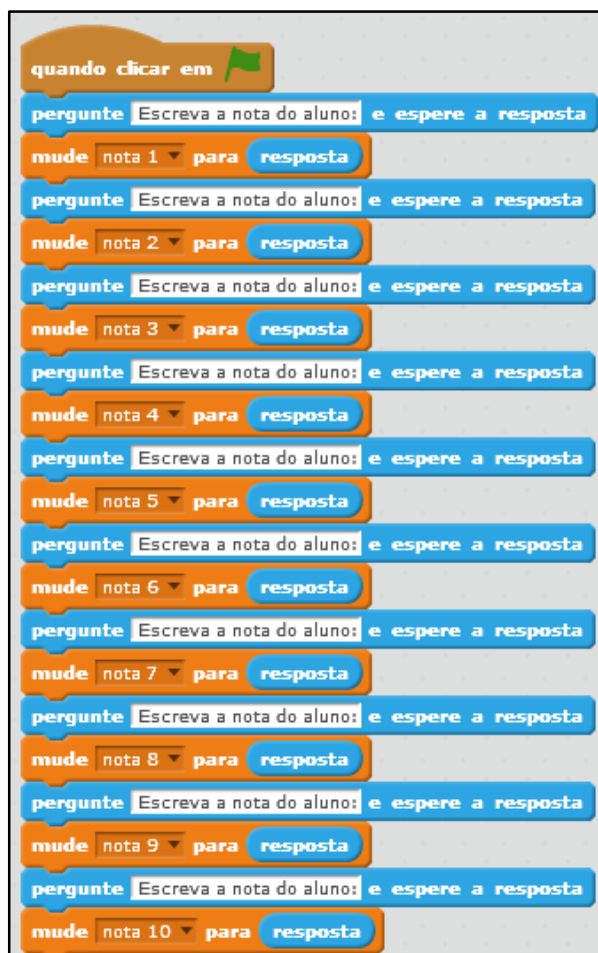
5 USO DE LISTAS

Lista é uma estrutura de dados referenciada por um nome que permite armazenar vários valores durante a execução do programa. Também podem ser chamadas de coleções de variáveis do mesmo tipo agrupadas e acessadas por um único identificador. No Scratch, elas podem ser utilizadas para armazenar números e/ou palavras e podem ser criadas na mesma seção de Variáveis.

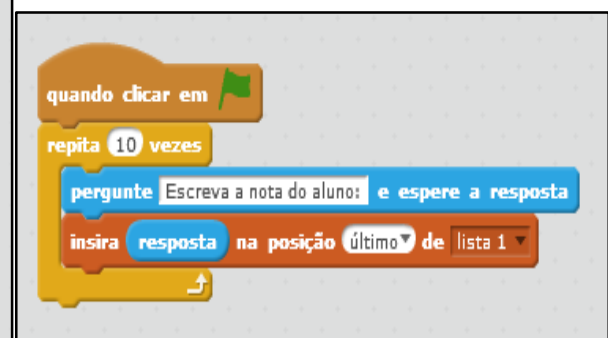
Listas são importantes para armazenar dados que serão processados no futuro e para evitar desperdício de blocos, ou seja, se quiséssemos salvar 10 notas de alunos, sem uma lista seria necessário criar 10 variáveis para as notas dos 10 alunos.

Ex: Lista de compras!

Importante: Fale sobre o funcionamento da lista, sobre como acessar seus itens, no que isto difere das variáveis.



Sem uso de lista



Com uso de lista

5.1 Atividade 13 - LISTA DE MATÉRIAS

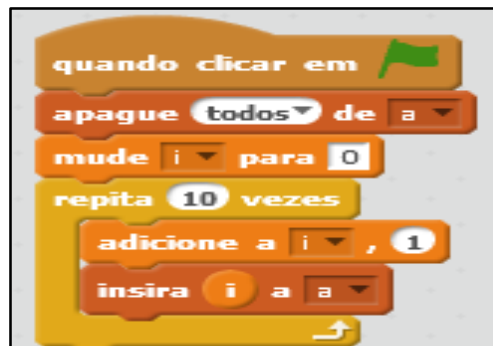
- ❑ Criar uma lista com o nome “Matérias”. Quando a bandeira verde for clicada, a lista deve ser limpa (apagar todos seus itens);
- ❑ O ator deverá perguntar o nome de uma matéria da escola e inserir a resposta na lista; Repetir isto 10 vezes e a seguir, o ator deverá informar que irá ler a lista;
- ❑ Dizer todos os itens da lista, um por um, por 1 segundo cada. Para isto:
 - Deve-se criar uma variável para controlar o índice da lista que será lido no momento, chame-a de “i” ou “índice”;
 - O ator deve dizer o item que está armazenado no **índice** atual da lista, e então somar 1 ao valor do **índice**;
 - Este processo deve ser repetido um número de vezes igual ao tamanho da lista.



Sugestão de comando para execução da atividade 13.

5.2 Atividade 14 - LISTA DE NÚMEROS

- ❑ Crie uma lista contendo os números de 1 a 10.
- ❑ Deve-se usar uma variável para controlar o número a ser inserido.



Sugestão de comando para execução da atividade 14.

5.3 Atividade 15 - INVERTENDO OS ELEMENTOS

- Crie duas listas, chamando-as de “Lista inicial” e “Invertida”. Quando a bandeira for clicada, as duas listas devem ser limpas;
- Insira 5 palavras informadas pelo usuário na Lista Inicial;
- Diga que vai preencher a lista Invertida (por 2 segundos);
- Em seguida, insira inversamente os elementos da Lista Inicial para a lista Invertida, de forma que o último elemento da primeira seja o primeiro elemento da segunda.
- Faça isto usando variável e esperando 1 segundo após cada inserção na lista Invertida;
- Ao final da atividade, peça para que os alunos leiam as listas e confirmem o resultado.

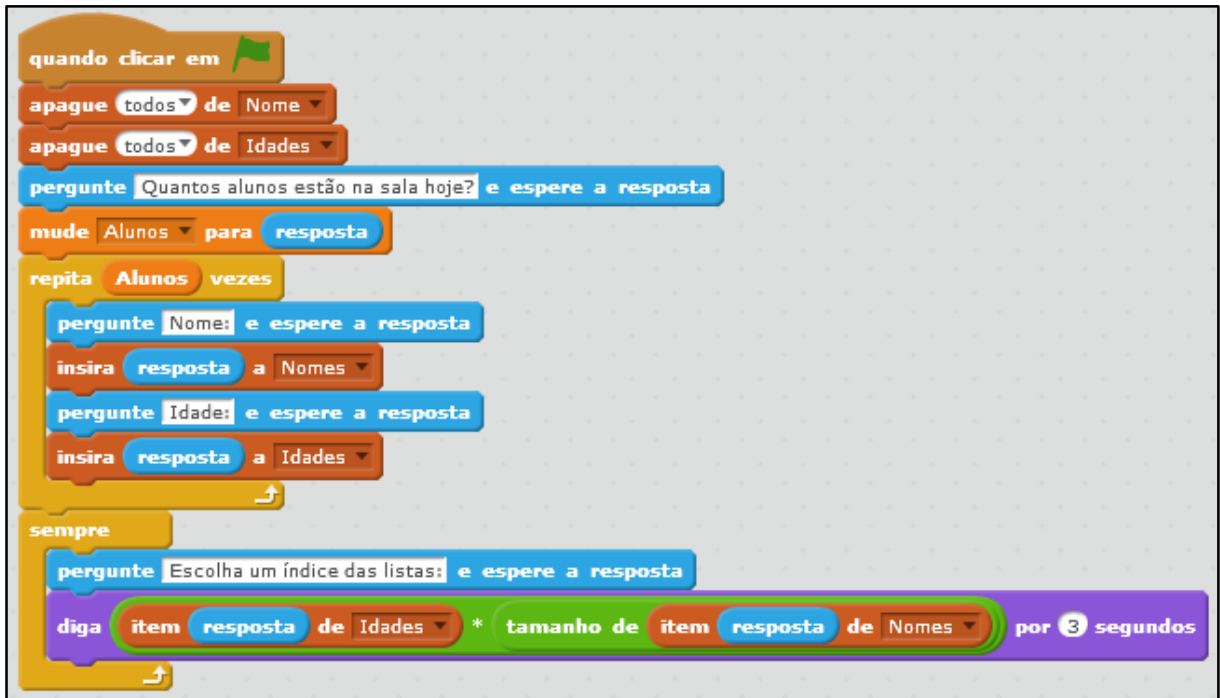


Sugestão de comando para execução da atividade 15.

5.4 Atividade 16 - NOME E IDADE

- Criar duas listas, uma chamada “Nome” e outra “Idade”;
- Perguntar quantos alunos estão na sala na aula atual;
- A lista NOME vai conter o nome dos colegas do aluno;

- ❑ A lista IDADE vai conter as idades referentes aos nome inseridos na outra lista, ou seja, a idade de seus colegas;
- ❑ A partir de agora o ator deve sempre perguntar um índice das listas e dizer, por 3 segundos, o resultado da seguinte multiplicação:
 - IDADE * Número de letras do NOME.



Sugestão de comando para execução da atividade 16.

5.5 Atividade 17 - LOTERIA DE PALAVRAS

- ❑ Crie uma lista para armazenar palavras (com o nome a sua escola) e uma variável chamada “sorteada”;
- ❑ Ao clicar na bandeira verde, a lista deve ser preenchida:
 - Pergunte ao usuário 5 palavras e as insira em posições aleatórias da lista;
 - Insira mais 5 palavras na lista (exemplos na imagem de sugestão);
- ❑ Faça com que o programa sorteie um dos nomes aleatoriamente da lista VETPALAVRAS e grave este em uma variável “UNICA”;
- ❑ Pergunte ao usuário qual a quantidade de letras contidas nesta palavra;
- ❑ Se o valor fornecido pelo usuário for correto, apresente uma mensagem parabenizando pelo acerto, caso contrário, acusando erro;

- ❑ O programa deverá repetir sempre, a menos que você clique no botão Parar Tudo(vermelho). Use a lista ÚNICA para manipular a palavra a ser sorteada na contagem da quantidade de suas letras.



Sugestão de comando para execução da atividade 17.

5.6 Atividade 18 - DE TRÁS PARA FRENTE

- ❑ Crie uma lista;
- ❑ Pergunte ao usuário que digite um número qualquer;
- ❑ Insira na lista os números inteiros de 1 até o número digitado pelo usuário, utilizando o repita;
- ❑ Faça o gatinho dizer os números contidos na lista de trás para frente, utilizando uma variável para manipular a posição dos números da lista;
- ❑ Ao final, o gatinho deve dizer o tamanho da lista.

- ❑ Escreva o esquema correspondente ao programa Scratch.



Sugestão de comando para execução da atividade 18.

OBS.: por causa das características da estrutura LISTA, a cada vez que o programa é executado, a lista aumenta. Assim, a cada rodada, primeiro ELIMINE a lista minha-lista (no menu a esquerda) e depois CRIE a lista minha-lista. Assim, o programa sempre vai começar com uma lista vazia.

- a) Modifique o segundo loop do programa para o gatinho dizer os elementos do início ao fim.



Sugestão de comando para execução da atividade 18a.

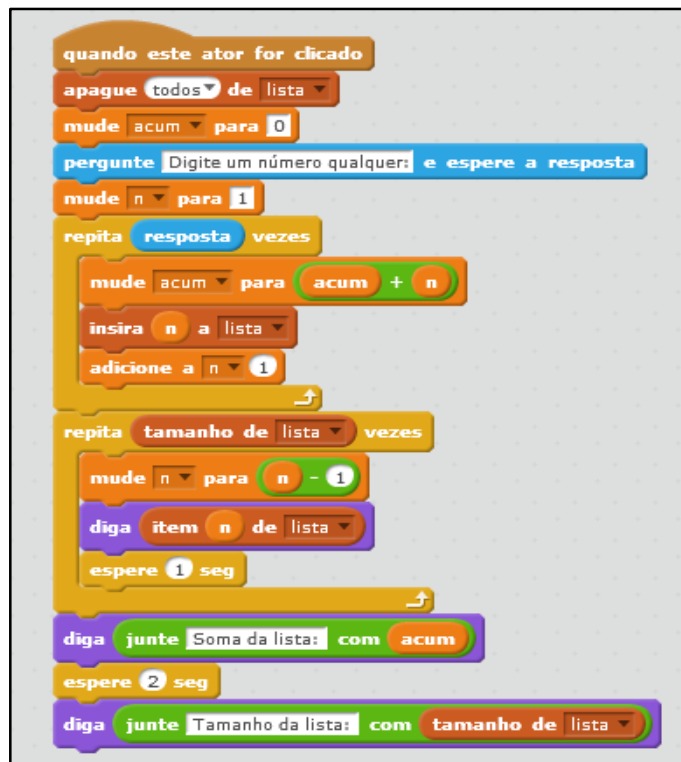
b) Modifique o programa para inserir apenas números pares dentro da lista, a partir de 2 até $2 * N$ (número digitado pelo usuário). O número informado pelo usuário irá determinar o tamanho que a lista terá.



Sugestão de comando para execução da atividade 18b.

c) Modifique o segundo loop do programa para o gatinho dizer a soma dos elementos da lista.

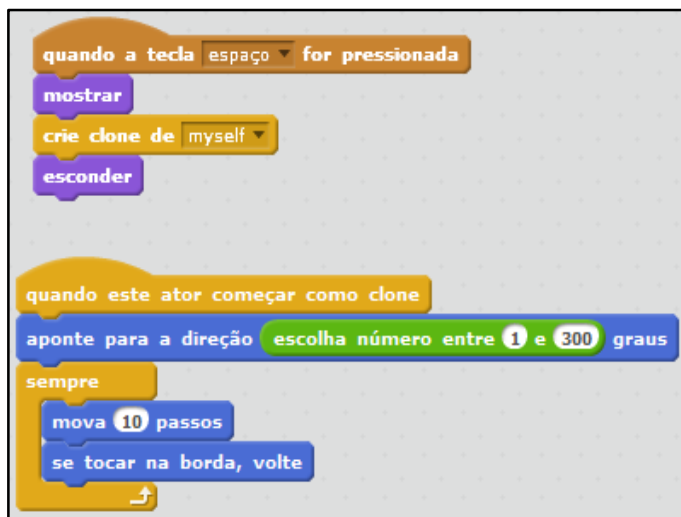
Dica: Para acompanhar os valores das variáveis, lembre-se de clicar nas caixinhas correspondentes.



Sugestão de comando para execução da atividade 18c.

5.7 Atividade 19 - CRIE CLONES

- Escolha um ator na biblioteca de atores;
- Crie clones do ator, quando clicado nele;
- Faça o clone apontar para uma direção qualquer entre 1 e 300 graus;
- Faça o clone se movimentar, respeitando a borda.



Sugestão de comando para execução da atividade 19.

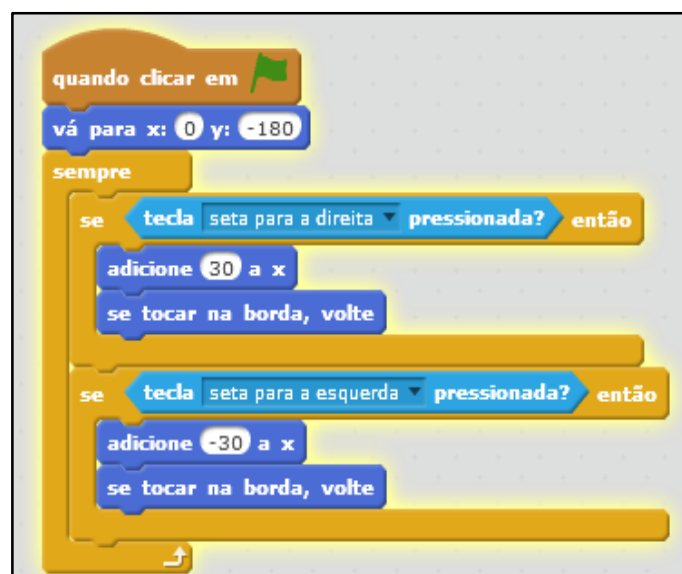
5.8 Atividade 20 - PEGUE A MAÇÃ

1. Criar um jogo onde:
 - maçãs aparecem em posições horizontais aleatórias na parte superior da tela do jogo,
 - em instantes variados, caem no chão.
2. O jogador deve mover o carrinho para pegar as maçãs antes que elas toquem o chão, sendo que cada uma delas vale 1 ponto.
3. A seguir, escolha um palco que possua uma árvore, desenhe um ator chamado "carrinho", escolha na biblioteca de atores um ator "maçã".



Dica: No Scratch 2.0 existe o recurso de clonagem. Como nesta atividade serão necessárias muitas maçãs, você poderá criar várias cópias do ator maçã, usando “criar clone”.

- Programação carrinho



Sugestão de comando para execução da atividade 20.

- Programação maçã



Sugestão de comando para execução da atividade 20.

Obs.: “Sprite1” é o carrinho; os blocos de som não devem ser inseridos caso o aluno não tenha um fone de ouvido. A expressão “ $y < -135$ ” se refere a maçã tocando o chão. Você pode testar outro valor para isto, como -160.



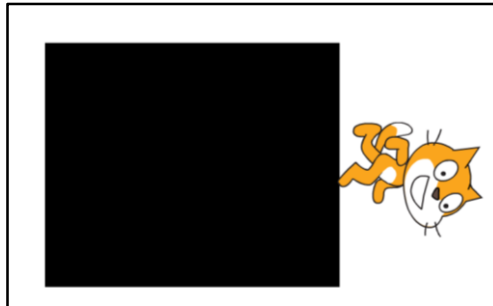
Sugestão de comando para execução da atividade 20.

5.9 DESAFIO 01

- construção do quadrado e inserção do gatinho: 1 ponto;
- movimentação e rotação ao redor do quadrado: 4 pontos.

PONTUAÇÃO TOTAL: 5 pontos.

O gatinho do Scratch corre sobre o contorno de um quadrado, conforme mostrado na figura abaixo:



Desafio:

- 1- **Desenhar** um quadrado no palco, conforme descrição feita no enunciado.
- 2- Quando a bandeira for clicada, o gato deve começar a correr ao redor do quadrado, pisando nas suas linhas.
- 3- Não use os blocos “deslize até...”, “vá para...” ou comandos diretos de mudar posição. Deve-se fazer com que o ator ande usando o bloco “mova () passos” e troque de fantasia sempre que andar.



Sugestão de comando para execução do Desafio 1.

5.10 Atividade 21 - FURAR BALÃO

1. Crie dois atores: furador e balão



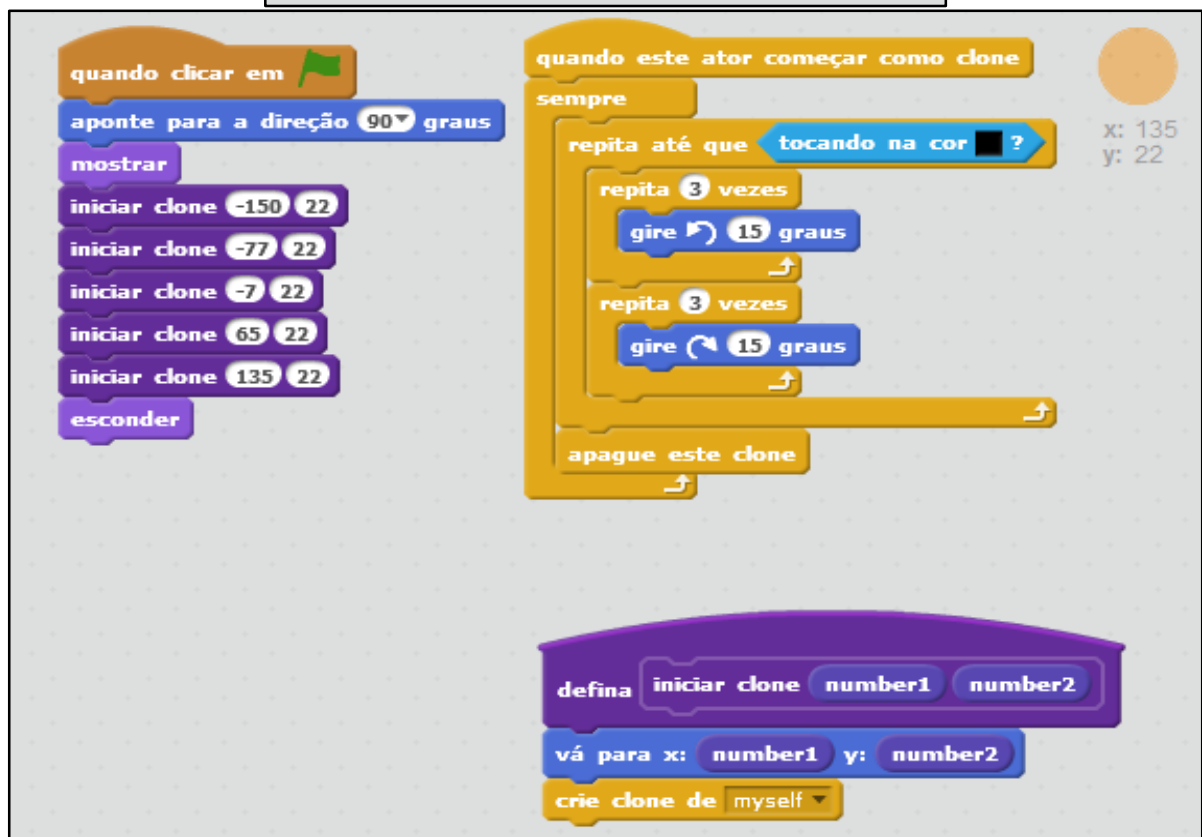
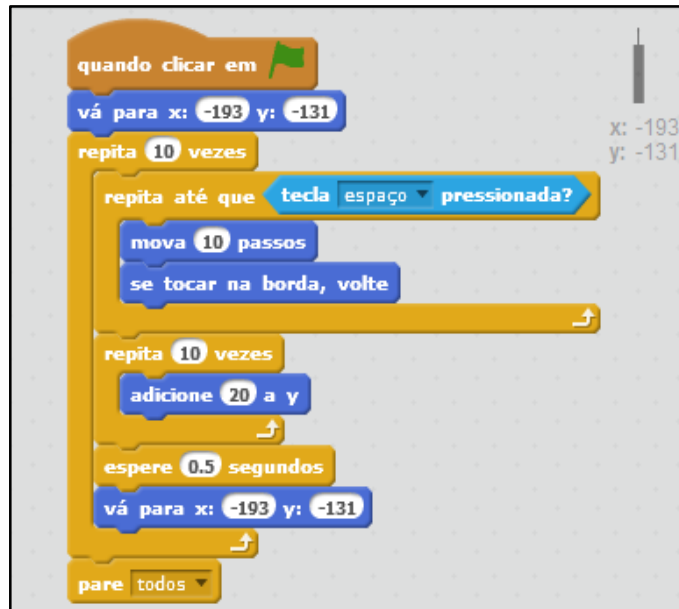
Furador



balão

2. Faça o ator furador se mover em toda a tela (horizontal e vertical) até que seja dado um comando de parada.
3. A posição inicial do ator furador é no inferior da tela.
4. Crie 5 clones do ator balão posicionados na parte superior da tela, um ao lado do outro. (Dica: o conceito de mais blocos poderá ser usado nesta atividade),

5. Quando o furador tocar o balão, este deverá trocar de traje e desaparecer.
6. Escrever uma mensagem na tela quando os balões forem todos furados.
7. Utilize o cronômetro e estipule um tempo máximo para furar todos os balões.



Sugestão de comando para execução da atividade 21.

5.11 DESAFIO 02

- entrada dos valores da base e da altura: 1 ponto;
- construção do retângulo baseado nas medidas: 3 pontos;
- mostrar o cálculo da área, juntamente com os valores das medidas: 1 ponto.

PONTUAÇÃO TOTAL: 5 pontos.

Um estudante está trabalhando com a área do retângulo, e sabe que a fórmula é $\text{área} = \text{base} \times \text{altura}$. O professor dele solicita que um retângulo seja desenhado na tela após fornecer, via teclado, as medidas da base e da altura. Além disso, ele pede para o aluno calcular e exibir na tela a área desse retângulo.

Desafio:

- 1- Fornecer **valores numéricos** para a base e para a altura. Faça com que o ator **desenhe** um retângulo qualquer utilizando esses valores.
- 2- **Mostre**, na tela, esses dois valores.
- 3- Apresente automaticamente o **cálculo da área**.



Sugestão de comando para execução do Desafio 2.

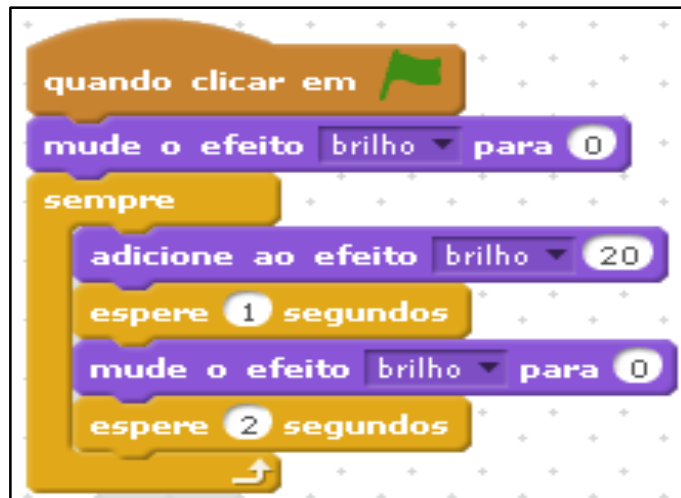
5.12 Atividade 24 - FOGOS DE ARTIFÍCIO

- Criar uma animação com o objetivo de demonstrar a explosão de fogos de artifício no céu.
 - Os fogos de artifício irão explodir em instantes aleatórios, gerando centelhas que cairão como se estivessem sob a ação da gravidade e que desaparecerão lentamente com o tempo.
- a. Crie ou insira um pano de fundo noite, um ator cidade e um ator rojão com 11 fantasias (4 rojões com cores diferentes e 7 fagulhas diferentes), conforme modelo da figura abaixo.



Sugestão de comando para execução da atividade 24a.

b. Na programação do palco, crie um efeito de brilho, dando a impressão com que cada vez que um rojão estoura no ar, este dá um clarão no céu.

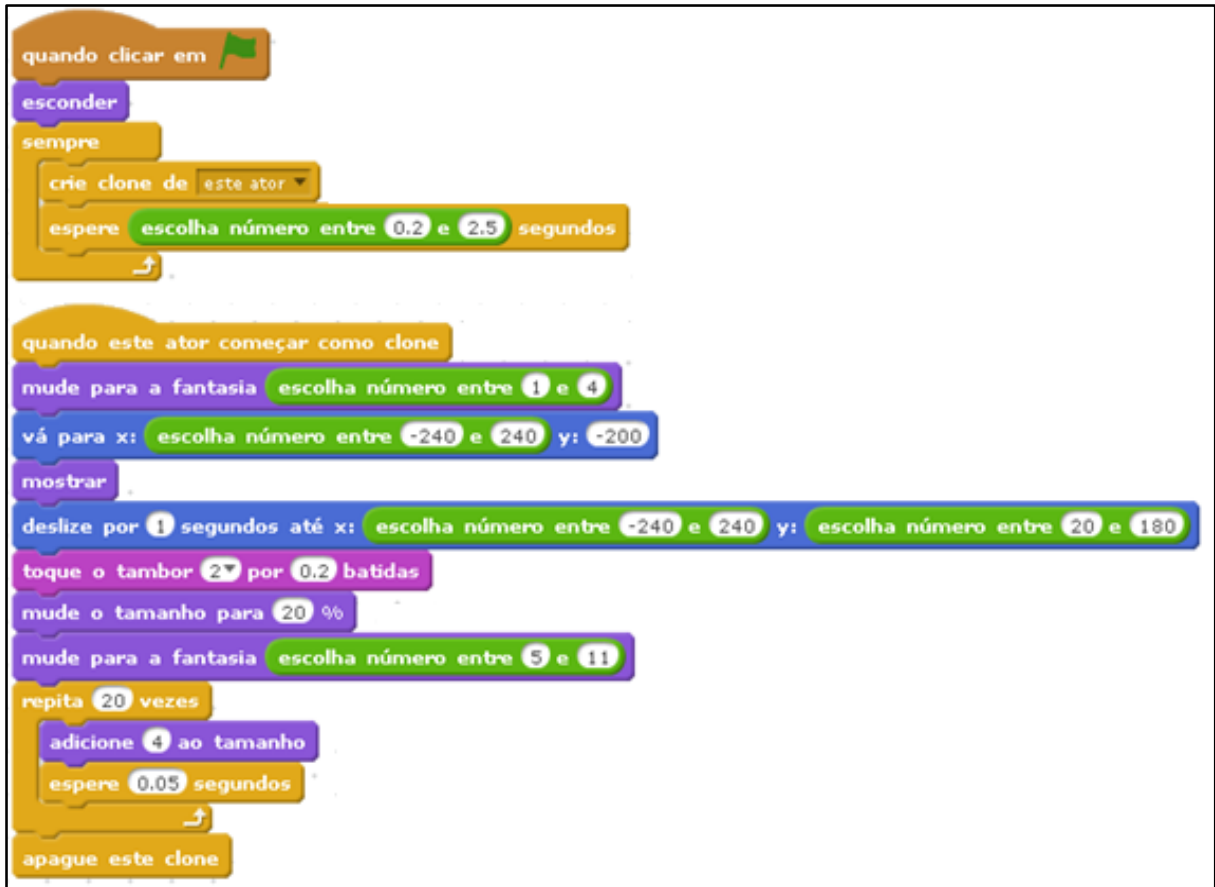


Sugestão de comando para execução da atividade 24b.

c. Programe, agora, o rojão.

- crie clone do rojão.
- ao criar o clone, escolha, de forma aleatória, uma das 4 primeiras fantasias.
- faça o clone se movimentar e se posicionar num x e y aleatório.
- escolha uma das 7 fantasias que representam a fagulha e mude o traje.

Dica: antes de mostrar o código, verifique se a ordem dos trajes deste personagem/ator são os quatro primeiros com os trajes das "fagulhas", e o restante com os trajes dos "rojões", se não o funcionamento do código ficará errado.



Sugestão de comando para execução da atividade 24c.

5.13 DESAFIO 03

- construção da escada e a inserção do macaco: 1 ponto;
- construção de degraus da escada com o mesmo espaçamento (passos do macaco): 1 ponto;
- movimentação do macaco: 1 ponto;
- variável que controla os passos: 2 pontos.

PONTUAÇÃO TOTAL: 5 pontos.

O macaco Kako deve subir uma escada com 18 degraus e alcançar o cacho de banana, que está no topo desta. Só que a escada está muito lisa e Kako tem dificuldade de subir. A cada passo, Kako sobe três degraus e desliza dois.

Desafio:

- 1- **Construir** o objeto escada, conforme a descrição.
- 2- Mostrar Kako **subindo** a escada, conforme o critério descrito.
- 3- Mostrar o total de **passos** que Kako levou para chegar ao topo da escada.



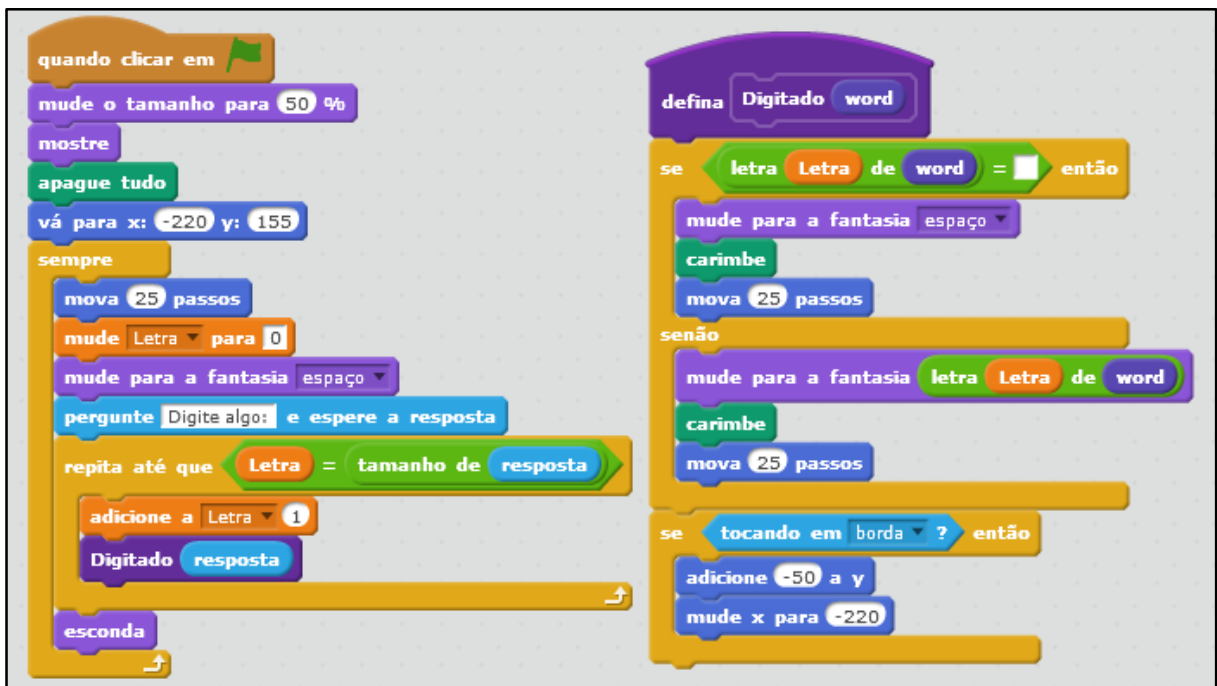
Sugestão de comando para execução do Desafio 3.

5.14 Atividade 25 - TYPE

- 1- Importe da biblioteca um ator letra
- 2- Importe, para este ator, fantasias de todas as letras do alfabeto, todos os Algarismos, alguns símbolos de pontuação e crie uma fantasia (em branco) chamada espaço.
- 3- Programe o ator obedecendo o que segue:
 - posicione o ator no canto superior esquerdo da tela;
 - pergunte uma palavra ou um número;
 - carimbe na tela todas as letras ou números digitados na tela;
 - o processo de perguntar e carimbar deverá ser repetido sempre até que um comando seja dado para parar;

- as palavras ou números não podem se sobrepor.

Dica: Crie um bloco para carimbar as palavras ou números na tela.



Sugestão de comando para execução da atividade 25.

5.15 DESAFIO 04

A assinatura digital geométrica de um número formado por dígitos diferentes de 0 é uma sequência de segmentos.

Os segmentos são traçados a partir do ponto $X = 0$ e $Y = 0$ seguindo a sequência de movimentos: **à direita, para baixo, à esquerda, para cima, à direita** e assim por diante (sentido horário).

O tamanho de um segmento depende do valor do dígito: para traçar o segmento que representa: o dígito 1 - mova 10 passos; para o dígito 2 - mova 20 passos; e assim por diante.

Exemplo: se quisermos obter a assinatura do número 35214, movemos 30 passos à direita, 50 para baixo, 20 à esquerda, 10 para cima e 40 à direita.

Desafio:

- Desloque um ator para a posição inicial;
- Desenhe a assinatura geométrica de um número fornecido por uma pergunta.
- Mude a cor da caneta a cada segmento.

Critérios para avaliação:

- Deslocamento para posição inicial: 5 pontos;
- Obedecer ao sequenciamento dos movimentos: 20 pontos;
- Tamanho do segmento: 5 pontos;
- Fornecimento de um número pelo usuário: 10 pontos;
- Separação do número em dígitos: 50 pontos;
- Mudança da cor da caneta: 10 pontos;

PONTUAÇÃO TOTAL: 100 pontos

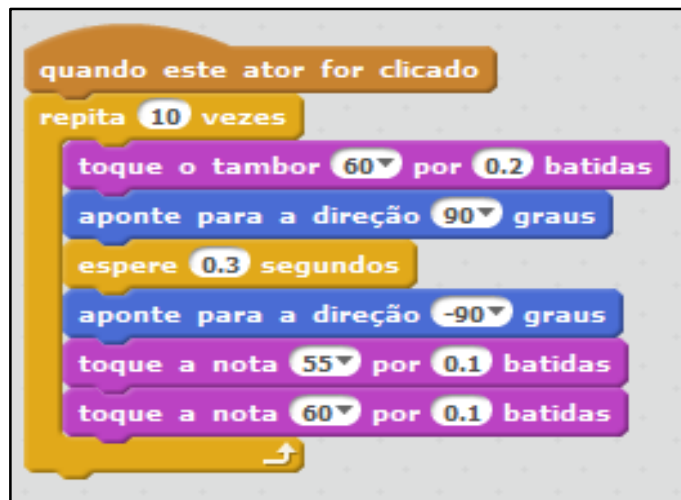


Sugestão de comando para execução do Desafio 4.

5.16 Atividade 26 - MÚSICA

- Insira o ator **cat2**;
- Faça este ator dançar, alternando a direção entre 90 ou -90;

- ❑ Faça tocar o tambor (60 por 0.2 batidas), e logo outras duas notas em sequência (55 por 0.1 batida e 60 por 0.1 batida).



Sugestão de comando para execução da atividade 26.

5.17 Atividade 27 - PARTITURAS

Essa tarefa consiste em tocar as notas da escala de natural completa de Dó, subindo a pauta em ordem crescente e voltando em ordem decrescente, assim:

dó-ré-mi-fá-sol-lá-si-dó-si-lá-sol-fá-mi-ré-dó.

1. Cada nota é representada por uma bola colorida que, quando tocada tem a cor alterada.
2. A pauta é formada por 5 linhas e 4 espaços. Cada espaço e linha corresponde a uma nota (vamos considerar a ordenação em clave de sol).

Dica: Existe uma notação internacional que define uma letra para cada nota, assim: C-dó; D-ré; E-mi; F-fá; G-sol; A-lá; B-si.

3. Utilize as ferramentas de som do Scratch e programe isso.
4. Construa a pauta mostrada na figura abaixo.



Dica: O que, na verdade, vai acontecer é que você implementará essa imagem abaixo, tocando os sons correspondentes.

No jogo os tijolos são alinhados no topo da tela.

A bola se movimenta pela tela, rebatendo nas paredes laterais e superiores da tela.

Quando um tijolo é atingido pela bola, a bola toca na parte inferior da tela. Para prevenir que isso aconteça, o jogador move uma raquete para rebater a bola para cima mantendo-a no jogo.

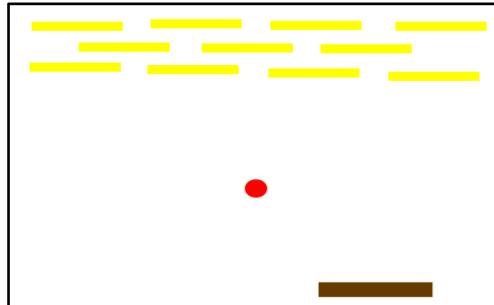


Imagem do jogo Breakout

Desafio:

- a) Insira um pano de fundo.
- b) Desenhe os atores que fazem parte do jogo. Os atores são denominados por:
 - i. raquete - retângulo marrom;
 - ii. tijolos - retângulos amarelos;
 - iii. bola - círculo vermelho.
- c) Faça a programação dos atores:
 - i. raquete - movimentação para direita e esquerda, respeitando as bordas;
 - ii. tijolos - posicionamento de pelo menos 11 tijolos e o desaparecimento quando a bola tocá-los;
 - iii. bola - movimentação automática, número de vidas (mínimo 3), controle das bordas, contato com o tijolo;
- d) Faça a finalização do jogo com dois panos de fundo (venceu e game over).

Critérios para avaliação:

- Inserir um pano de fundo (1 cenário): 1 ponto;
- Desenhar os atores - raquete, tijolos (mínimo 11) e bola: 4 pontos;
- Programar a raquete:
 - movimentação: 10 pontos;
 - respeitar bordas: 10 pontos;
- Programar os tijolos:

- posição de cada tijolo: 10 pontos;
- desaparecimento de cada tijolo quando tocado pela bola: 15 pontos;
- Programar a bola:
 - movimentação automática: 15 pontos;
 - número de vidas (mínimo 3): 5 pontos;
- controle das bordas: 5 pontos;
- contato com tijolo: 15 pontos;
- Finalização do jogo com dois planos de fundo:
 - venceu: 5 pontos;
 - game over: 5 pontos;

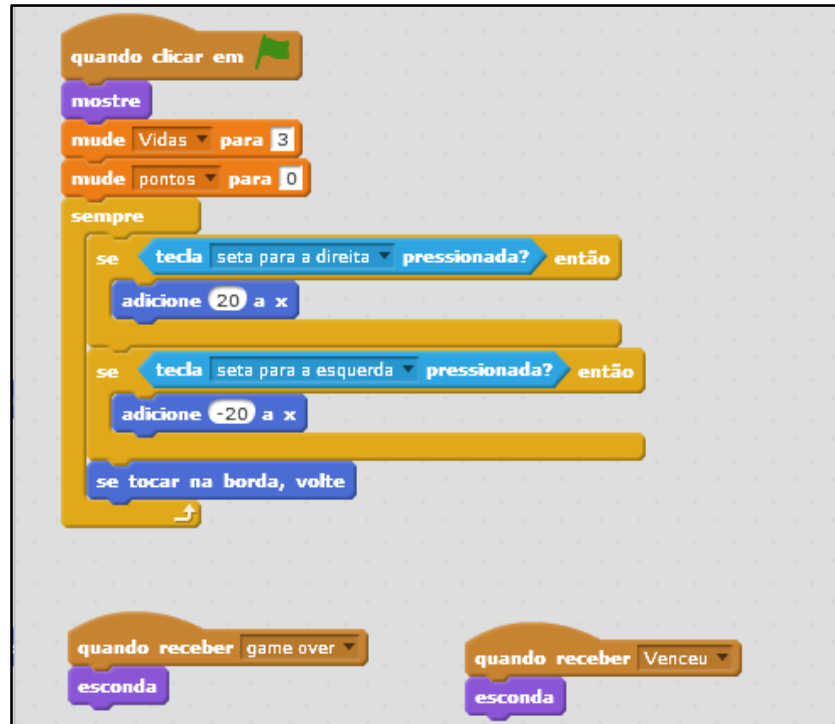
PONTUAÇÃO TOTAL: 100 pontos

- Programação de uma barrinha amarela, exemplo:



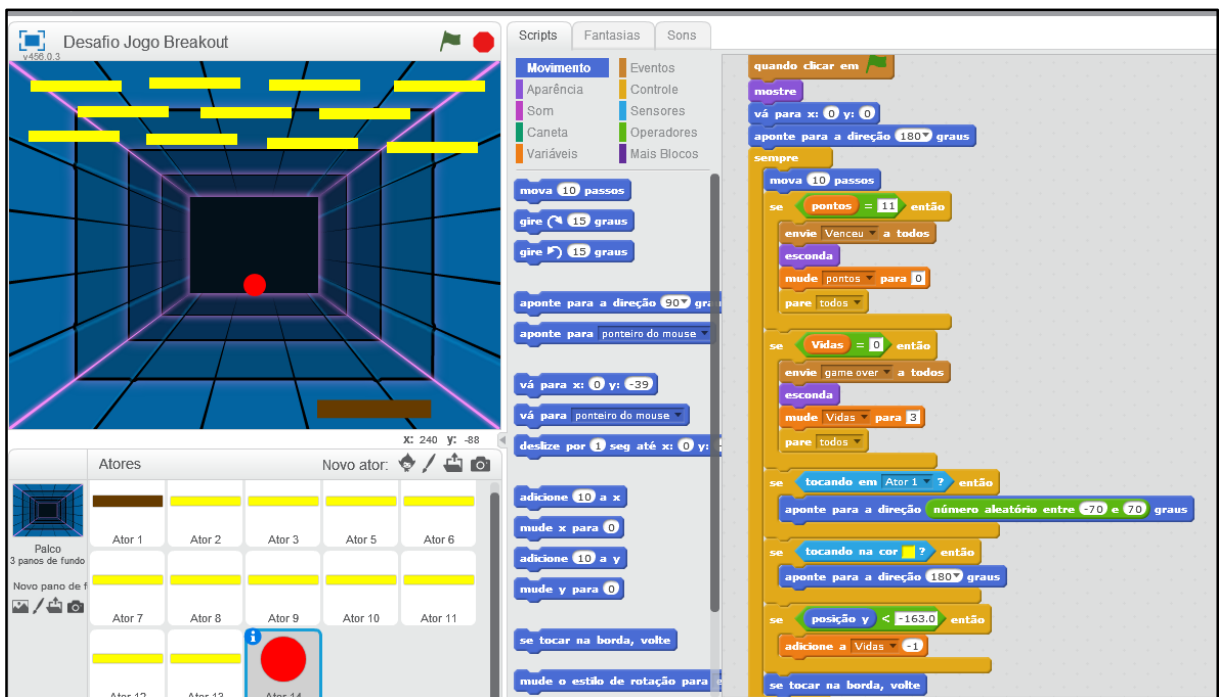
Sugestão de comando para execução do Desafio 5.

- Programação barrinha inferior:



Sugestão de comando para execução do Desafio 5.

- Programação da bolinha:




Sugestão de comando para execução do Desafio 5.

6 ATIVIDADES EXTRAS

6.1 Atividade 1 - CRIAR UM ATOR

Exclua o ator gatinho;


Crie um ator retangular usando o editor de pintura .


Salvar atividade em um arquivo.


6.2 Atividade 2 - INSERIR SOM NO ATOR

Criar um arquivo novo (vai aparecer o ator gatinho).

Excluir o ator gatinho.

Inserir um objeto do arquivo .

Inserir um segundo objeto do arquivo .

Clicar sobre um dos objetos e inserir um som usando .

Salvar atividade em um arquivo.

6.3 Atividade 3 - MUDAR A FANTASIA

Crie um novo arquivo;

Exclua o ator gatinho;

Insira um palco a sua escolha da pasta “natureza”;


Escolha um ator que possibilite a mudança de fantasia e o insira no palco;


Faça a mudança da fantasia do ator escolhido.

6.4 Atividade 4 - IMPORTAR ATORES

Criar um arquivo novo.

Salvar imagens da internet em um arquivo do computador.

Importar a imagem salva no arquivo do computador usando .

Usando o botão crescer objeto e encolher objeto (dois últimos botões de ) aumente e diminua o tamanho da figura.

Inclua um palco de acordo com a figura escolhida.

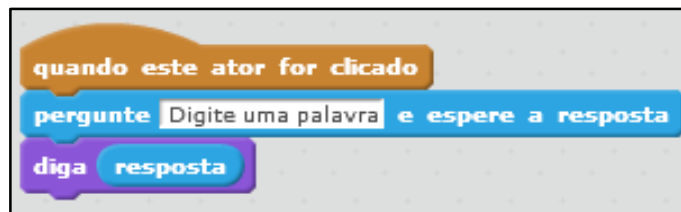
Finalize a atividade deixando o palco e as figuras esteticamente apresentáveis.

6.5 Atividade 5 - PERGUNTA

Insira um ator “rato”;

Quando clicado, o ator pergunta ao usuário o fornecimento de uma palavra qualquer;

Em seguida, o ator deverá mostrar a palavra fornecida.



Sugestão de comando para execução da atividade 5.

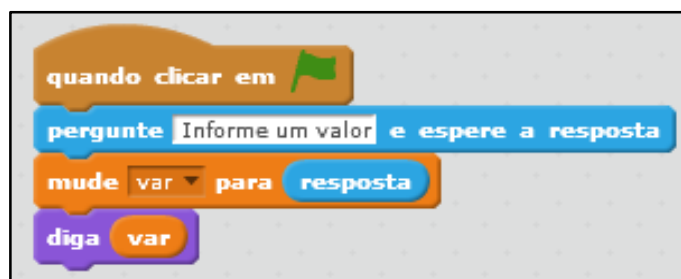
6.6 Atividade 6 - VARIÁVEL RESPOSTA

Insira um ator qualquer;

Crie uma variável chamada Valor;

Quando a bandeira verde for clicada, o ator pergunta ao usuário a digitação de um número;

Esse número é guardado na variável Valor e deverá ser mostrado a partir desta variável.



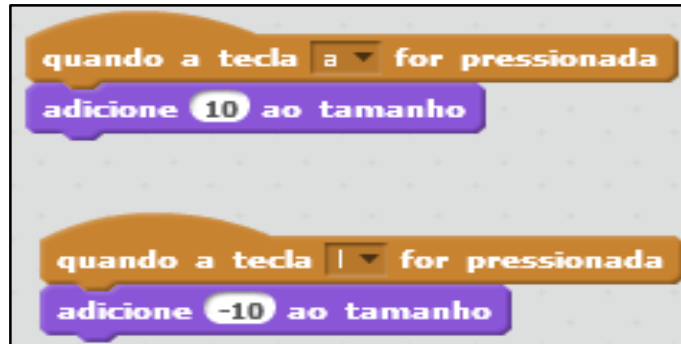
Sugestão de comando para execução da atividade 6.

6.7 Atividade 7 - TAMANHO DO ATOR

Insira um ator qualquer;

O ator deve ter o seu tamanho alterado à proporção de 10 e -10;

Quando a tecla A for clicada o ator irá diminuir de tamanho e quando a tecla L for clicada irá aumentá-lo.



Sugestão de comando para execução da atividade 7.

6.8 Atividade 8 - RAIZ QUADRADA

Insira um ator qualquer;

Quando o ator for clicado perguntará ao usuário que digite um número;

Logo, o ator diz o cálculo da raiz quadrada deste número.



Sugestão de comando para execução da atividade 8.

6.9 Atividade 9 - DIA E NOITE

Escolha duas imagens da internet, uma que represente uma lâmpada acesa e outra apagada;

Faça o upload das imagens dentro da aba fantasias;

Escolha dois palcos que representem a noite e o dia;

Quando o objeto lâmpada for clicado o cenário (palco) deverá mudar;

Para apresentar o cenário NOITE, a lâmpada deve ser apagada. Para apresentar DIA, a lâmpada deve ser acesa.



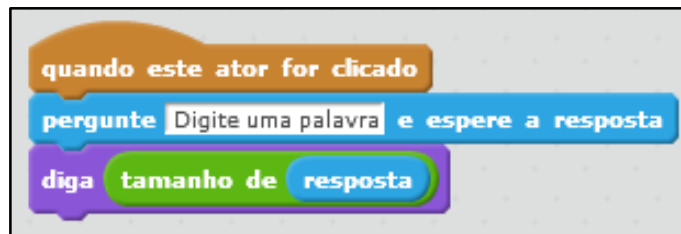
Sugestão de comando para execução da atividade 9.

6.10 Atividade 10 - TAMANHO DA PALAVRA

Insira um objeto qualquer no palco.

Esse objeto, quando clicado, solicita a digitação de uma palavra qualquer.

Logo, o mesmo objeto diz o tamanho da palavra digitada pelo usuário;



Sugestão de comando para execução da atividade 10.

6.11 Atividade 11 - ADIÇÃO DE VARIÁVEIS

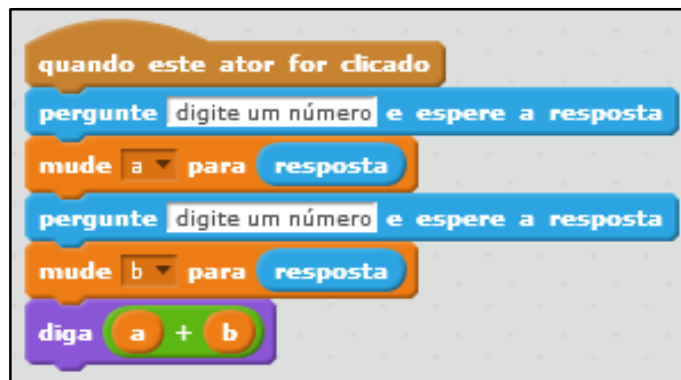
Crie duas variáveis: A e B;

Insira o ator FROG (sapo);

Quando o ator for clicado será perguntado ao usuário dois valores em sequência;

Um desses valores será atribuído a variável A e outro ao B;

Logo o ator deve dizer o resultado da soma das variáveis.



Sugestão de comando para execução da atividade 11.

6.12 Atividade 12 - LABIRINTO

Inserir quatro atores:

Gatinho, Chegada, Saída/Partida e Corredor (trajeto é desenhado em diagonal);

O Gatinho deve iniciar o programa sempre perto do objeto Saída;

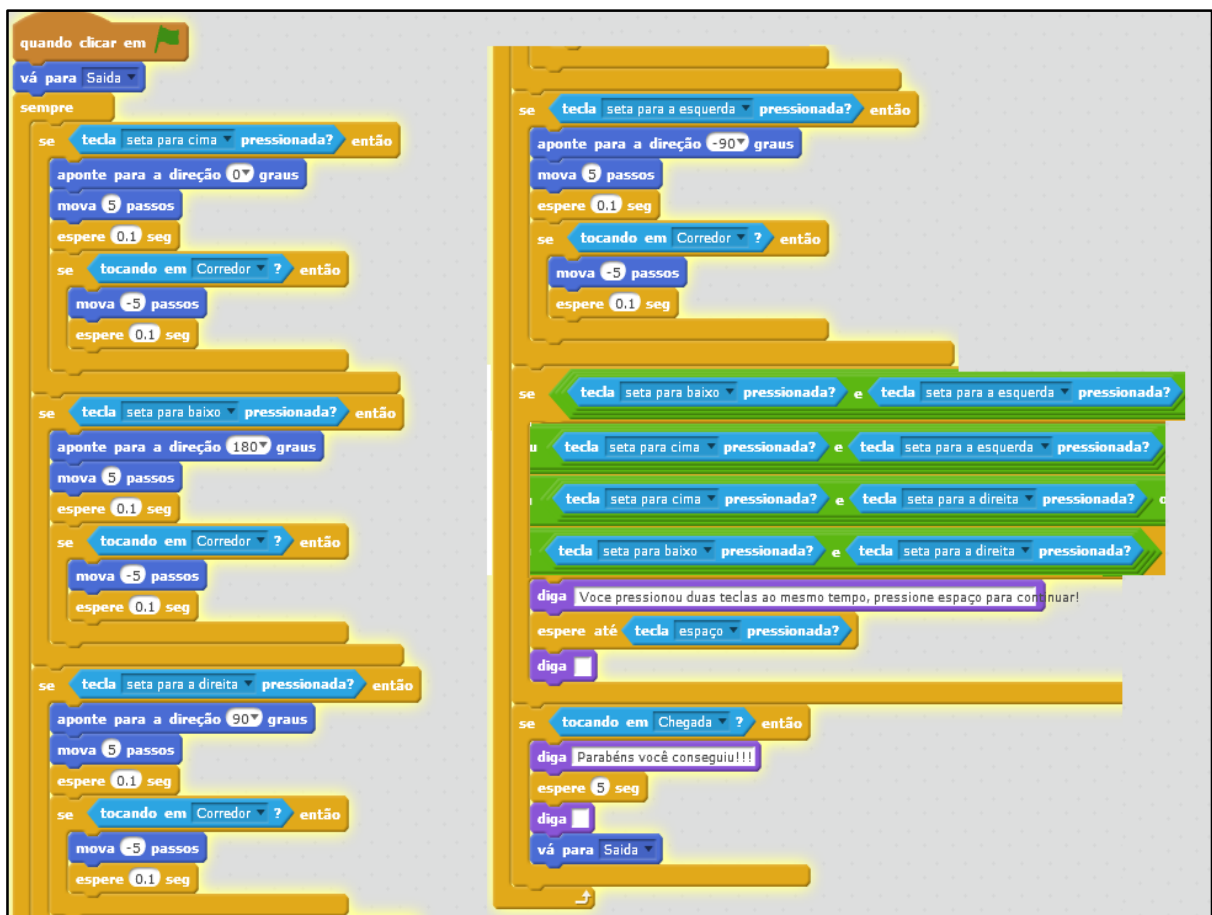
O Gatinho deve se movimentar utilizando as teclas direita, esquerda, para cima e para baixo do teclado;

Controle o Gato para que ele não exceda os limites do caminho a ser percorrido, ou seja, quando o objeto encostar no corredor ele deve voltar um passo;

O objeto não pode mover-se se você apertar duas teclas simultaneamente (deve ser pressionada a tecla espaço para prosseguir);

Quando o objeto encostar na Chegada o gato deve dizer: “Parabéns, você conseguiu!”

Ao alcançar a chegada, o ator volta ao ponto de partida.



Sugestão de comando para execução da atividade 12.

6.13 Atividade 13 - MULTIPLICAÇÃO DE VARIÁVEIS

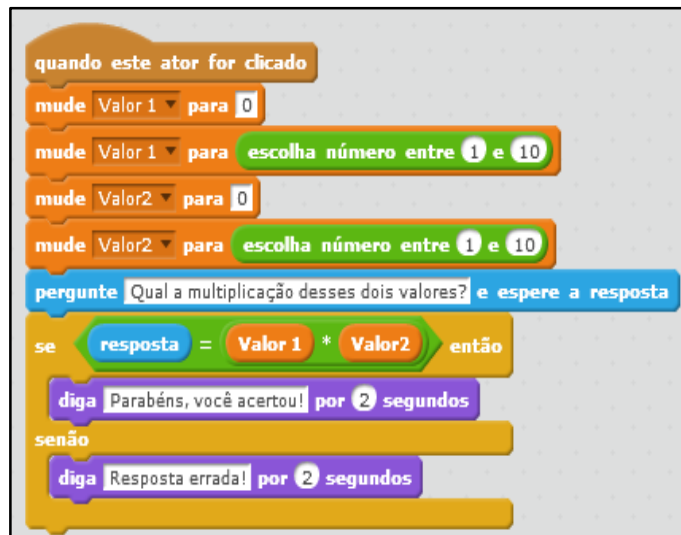
Crie ou insira um botão

Crie duas variáveis que recebem os seguintes nomes: Valor 1 e Valor 2

Programa o seguinte:

Quando o botão é clicado, ele sorteia um número aleatório e insere esses valores às variáveis;

- a animação deve perguntar a resposta da multiplicação entre os dois valores sorteados;
- se a resposta estiver correta, mostre a mensagem “Resposta certa”. Do contrário, mostre “Resposta errada”. Utilize as duas variáveis criadas para manipular esse cálculo e indicar a situação da resposta: se está correta ou não.



Sugestão de comando para execução da atividade 13.

6.14 Atividade 14 - MATEMÁTICA DIVERTIDA

Pergunte ao usuário que digite dois números em sequência;

Crie duas variáveis A e B;

Insira os números informados pelo usuário, um em cada variável;

Faça o ator dizer a soma das variáveis na tela:

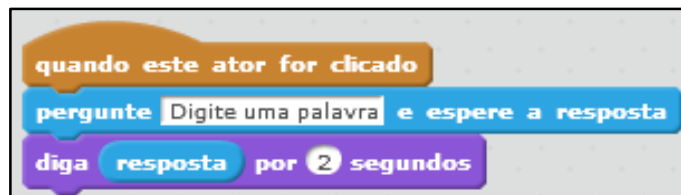
Agora mude o programa para fazer outras operações: multiplicar, dividir e subtrair.



Sugestão de comando para execução da atividade 14.

6.15 Atividade 15 - VOCABULÁRIO

Insira um ator “rato”. Quando clicado, ele solicita o fornecimento de uma palavra qualquer. Em seguida, ele mostra a palavra fornecida.



Sugestão de comando para execução da atividade 15.

6.16 Atividade 16 - ANÃO OU GIGANTE

Insira um ator. Ele deve ter o seu tamanho alterado à proporção de 10 e -10. A tecla A é utilizada para a diminuição do tamanho. A tecla L para aumentá-lo.

6.17 Atividade 17 - TOCA E DANÇA

Insira um objeto. Faça este ator dançar. Ele dança alternando a direção entre 90 ou -90 ao toque de um tambor (60 por 0.2 batidas), e logo de duas notas em sequência (55 por 0.1 batida e 60 por 0.1 batida).

6.18 Atividade 18 - CARIMBANDO O TRAJE

Crie três botões: TROCA TRAJE, LIMPAR, CARIMBA. O botão TROCA TRAJE permite a troca de traje de um objeto desenhado. Esse objeto tem 3 trajes diferentes: um quadrado perfeito e 2 retângulos. A utilização dos três permite criar, no palco, um novo quadrado perfeito. Através do botão CARIMBA, carimbe no palco o traje desejado. O botão LIMPAR, permite que você limpe o desenho criado e reutilize o palco. A tecla A serve para você girar o objeto 90° para a direita. A tecla S para girar o objeto 90° para a esquerda. As setas do teclado servem para que você posicione o objeto no palco.

6.19 Atividade 19 - MAIOR, MENOR E IGUAL

Pergunte ao usuário que digite um número de entrada;

Crie uma variável “var” e inicialize seu valor em 1;

Repita os procedimentos abaixo n vezes, onde n é o número armazenado na variável “resposta” fornecida pelo usuário;

Use o “Se...então...senão” para verificar se o valor armazenado em “var” é maior que 5, se for faça o gatinho dizer “MIAU” por 2 segundos e andar “var”*100 passos (dica: adicione o comando que permite ao gatinho voltar quando atinge as bordas do cenário).

Se o valor armazenado em “var” for menor ou igual a 5, faça o gatinho trocar de traje;

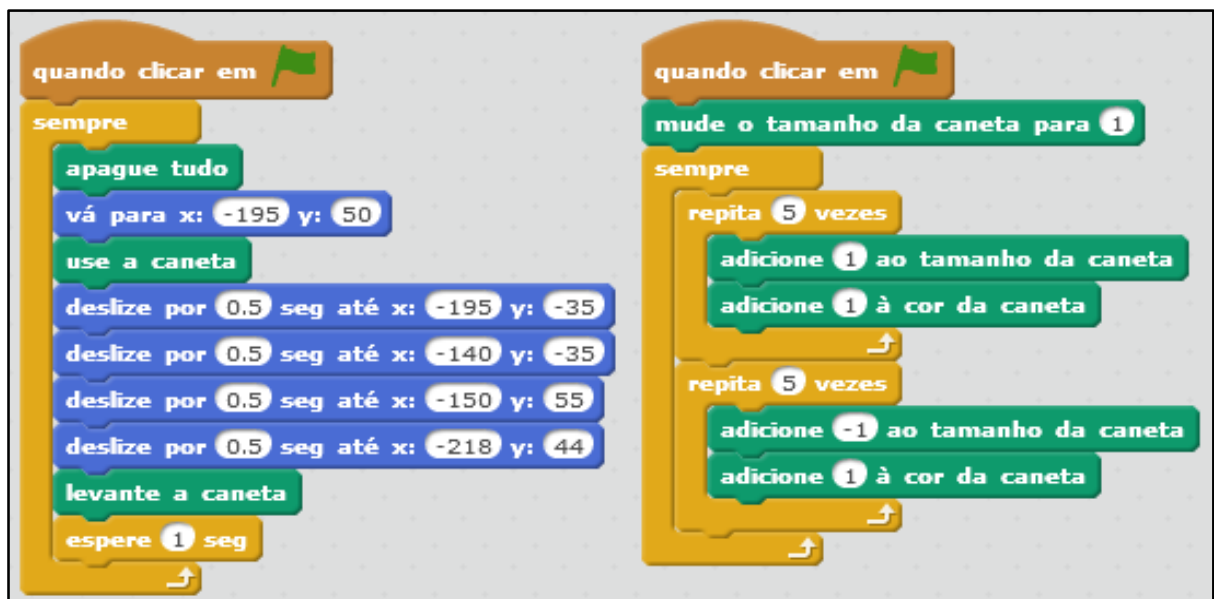
Por fim, adicione o valor de “var” em 1 e faça o gatinho esperar 2 segundos.



Sugestão de comando para execução da atividade 19.

6.20 Atividade 20 - ASSINATURA

- Use os comandos da caneta e deslize para criar sua assinatura no Scratch;
- O tamanho da caneta deve seguir variando de 1 até 6, 6 até 1 (iniciando em 1);
- A cor da caneta deve seguir sempre aumentando seu número em 1;
- A animação da assinatura deve ficar sempre executando.



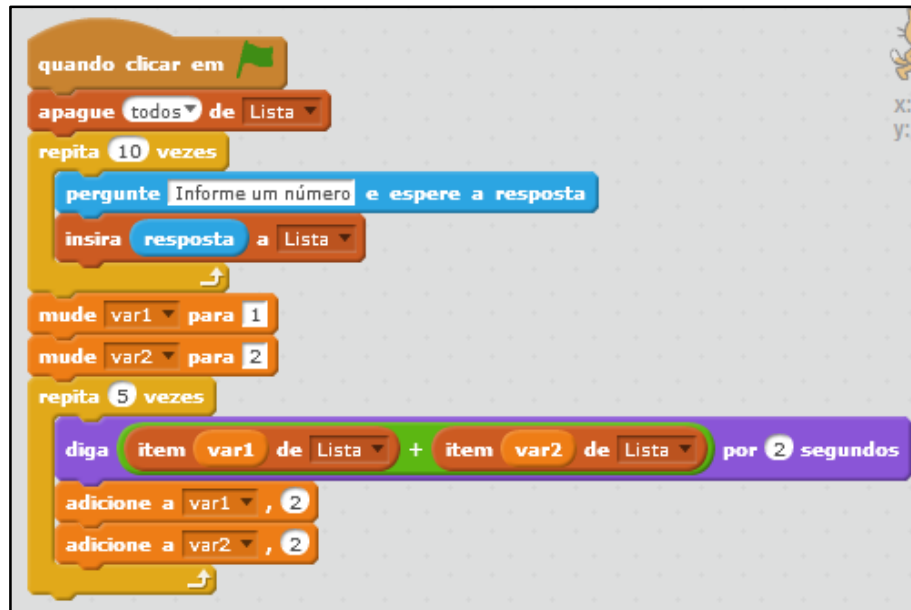
Sugestão de comando para execução da atividade 20.

6.21 Atividade 21 - ADIÇÃO COM O USO DE LISTA

Peça ao usuário que informe 10 números;

Criar uma lista contendo os 10 números informados pelo usuário;

Faça o gatinho dizer a soma do primeiro número da lista com o segundo, do terceiro com o quarto, assim sucessivamente até o número 10.



Sugestão de comando para execução da atividade extra 21.

7 ATIVIDADES A SEREM DESENVOLVIDAS

Nesta fase os alunos deverão criar seus próprios projetos, seja um game, jogo de tabuleiro, RPG, quiz ou animação, podendo acessar os exemplos no arquivo do Scratch. Esta atividade pode ser realizada em duplas (um com mais facilidade e outro com menos facilidade).

- **GAMES LITERÁRIOS:** Criação de games para divulgar uma obra literária, trabalhar com interpretação e criação a partir do estudo literário de uma obra existente e histórias de autoria dos alunos.
- **JOGOS OU GAMES DIGITAIS:** Criação de uma diversidade de jogos, com diferentes propostas, dentro das possibilidades de programação do Scratch. Os jogos digitais são muito atraentes para o público infantil e adolescente. É a oportunidade de reproduzir seus games favoritos ou de criar um jogo de sua autoria, saindo do papel de consumidor para o de produtor de games. Assim que os alunos compreendem a utilidade dos diversos comandos, iniciam a produção, partindo dos mais simples, para os mais complexos, conforme vão entendendo a lógica dos diversos comandos.
- **JOGOS DE TABULEIRO:** Programação de jogos com temas e desafios gerados a partir da criatividade do programador. A criação de jogos de tabuleiro utiliza outros recursos do Scratch que normalmente não são utilizados numa animação. Envolve o uso de sensores, operadores e, em alguns casos, variáveis.
- **RPG:** É a sigla inglesa de Role-Playing Game, que em português significa "jogo de interpretação de personagens". Consiste em um tipo de jogo no qual os jogadores desempenham o papel de um personagem em um cenário fictício. É um jogo diferente dos convencionais, pois não há ganhadores nem perdedores. Os jogos RPG estimulam a imaginação e o raciocínio lógico, desenvolvem a criatividade, o relacionamento interpessoal e a cooperação mútua. Cada jogo, também chamado de "sessão" ou "aventura", terá o enredo definido pelo jogador, mediante um determinado sistema, isto é, um conjunto de regras definidas para cada personagem. A decisão de cada um influenciará no destino do jogo.
- **CRIAÇÃO DE QUIZ INTERATIVO:** Pode ser uma oportunidade para o aluno criar desafios pensados a partir de um conhecimento construído. Para tanto, precisam elaborar perguntas e respostas possíveis para a interação das pessoas que

serão os usuários do jogo. Pode ser criado com informações pesquisadas em um projeto ou, até mesmo, após o estudo de determinados temas.

- **ANIMAÇÕES:** Criação de histórias animadas, com diálogos, efeitos sonoros e de movimento.

REFERÊNCIAS

EDUScratch. Site do Scratch para educadores. Disponível em:
<<http://eduscratch.dge.mec.pt/>>. Acesso em: 10 jun. 2017.

ORO, Neuza Terezinha; PAZINATO, Ariane Mileidi; TEIXEIRA, Adriano Canabarro.
Programação de Computadores na Educação: Um passo a passo utilizando o Scratch. Passo
Fundo: UPF Editora, 2016. 109 p.

SCRATCH. About Scratch (Scratch Documentation Site). Disponível em:
<<https://goo.gl/jQo3AU>>. Acesso em: 10 jun. 2017.