

ISSN 2316-7785

O SOFTWARE SCRATCH: UMA CONTRIBUIÇÃO PARA O ENSINO E A APRENDIZAGEM DA MATEMÁTICA

André Eduardo Ventorini¹
Universidade Federal de Santa Maria
ventoriniandre@gmail.com

Leandra Anversa Fioreze²
Universidade Federal de Santa Maria
leandra.fioreze@gmail.com

Resumo

O minicurso apresenta o software de programação scratch como recurso para subsidiar a aprendizagem da matemática relacionada à coordenadas cartesianas, ângulos, geometria analítica e geometria plana através da realização de atividades práticas e orientadas. A base teórica, tanto para analisar a construção dos conceitos matemáticos como para o planejamento das atividades utilizando um software de programação é sustentada pela teoria do Construcionismo de Seymour Papert. Busca-se, através deste minicurso, apresentar as noções básicas desta linguagem de programação, contribuir para o fortalecimento de novas práticas educativas nesta área do conhecimento e para as pesquisas relacionadas à construção de conceitos matemáticos através de programações com o scratch. Ao realizar as atividades, os praticantes desenvolvem habilidades tais como a construção de sequências de comandos que mostram a exploração de estruturas algébricas, na medida em que se incorpora nesta programação variáveis envolvendo números e operações matemáticas, a utilização de elementos da geometria analítica, na medida em que se manipula coordenadas cartesianas ao localizar ou mover personagens e objetos no traçado de retas e elementos da geometria plana, na medida em que se constroem figuras geométricas.

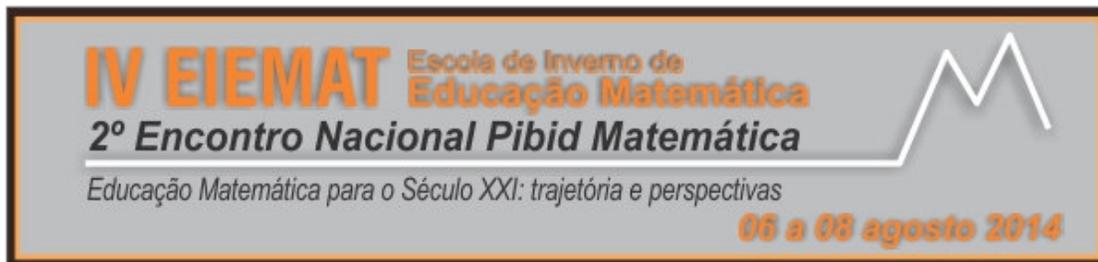
Palavras-chave: Scratch; programação; minicurso.

1. Introdução

Este minicurso caracteriza-se por trazer, além das noções básicas de programação do aplicativo em questão, uma abordagem diferente e inovadora do processo de construção de alguns conceitos matemáticos como plano cartesiano, ângulos, geometria analítica e geometria

¹ Professor de matemática da rede pública de ensino e aluno do Programa de Pós-Graduação em Educação Matemática e Ensino de Física da Universidade Federal de Santa Maria – UFSM.

² Professora Doutora em Informática na Educação pela Universidade Federal do Rio Grande do Sul – UFRGS e Adjunta da Universidade Federal de Santa Maria – UFSM.



plana através do software de programação Scratch, tendo como base a teoria de Seymour Papert. Pretende-se subsidiar, através da realização de atividades práticas e orientadas, a aprendizagem desses conceitos de maneira investigativa, criativa e desafiadora.

O minicurso destina-se a todos os profissionais interessados em experimentar novas práticas educativas, independentemente da modalidade ou série tanto do ensino fundamental como do ensino médio. Dentre os vários conteúdos de matemática, optou-se pela geometria analítica, geometria plana e ângulos devido à forma bastante tradicional e pouca desafiadora que este assunto vem sendo tratado em sala de aula.

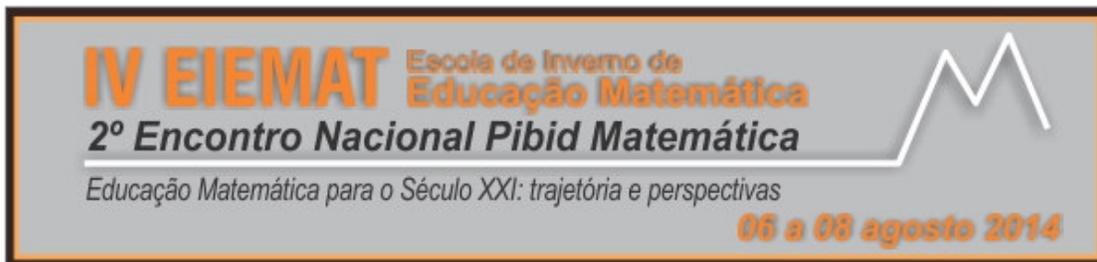
Considera-se que seja possível outra abordagem destes conteúdos, na qual os alunos possam ser os autores no processo da construção de conceitos, oportunizando assim uma aprendizagem significativa.

A metodologia proposta neste minicurso é uma sequência de atividades práticas e orientadas, partindo de situações introdutórias com a exploração dos comandos de programação do software scratch, para posteriormente utilizá-lo com fluência na construção de alguns conceitos matemáticos.

2. As Tecnologias de Informação e Comunicação (TIC) em Sala de Aula

Há algum tempo percebe-se uma tímida mudança na estrutura educacional do Brasil, quando se trata de inserções de tecnologias nos currículos escolares. Raras são as iniciativas de se tentar mudar este cenário onde a tecnologia clama e se apresenta para fazer parte dele. Os currículos das escolas não contemplam ainda de forma satisfatória esses novos recursos que podem contribuir de forma significativa no processo da construção do conhecimento.

Esse descompasso é um dos fatores desencadeadores da falta de motivação e evasão dos alunos nas escolas.



Será que estamos mesmo à espera que as crianças se mantenham passivas perante os currículos pré-digeridos do ensino básico, quando já exploram o saber contido nas auto-estradas da informação de todo o mundo e se abalaçaram a realizar projetos complexos, procurando por si próprias o conhecimento e os conselhos de que necessitaram para os pôr em prática? (PAPERT, 1997, p. 226).

Para Valente (1993), o computador não é um instrumento que ensina o aprendiz, mas ferramenta com a qual o aluno desenvolve algo, e, portanto, a aprendizagem ocorre pelo fato de estar executando uma tarefa por meio do computador.

Sob esse olhar, a educação precisa andar de mãos dadas com as novas tecnologias que fazem parte do quotidiano das pessoas e de uma forma particular, dos alunos que frequentam os bancos escolares.

2.1 O Software Scratch: programação voltada a objetos

Scratch é um ambiente produzido pelo Lifelong Kindergarten Group do Massachusetts Institute of Technology/MIT Media Lab. O software foi disponibilizado em Maio de 2007 para download em www.scratch.mit.edu. Ambos, o web site e o software, possuem versões em português.

Através do software scratch é possível trabalhar os seguintes conceitos específicos de programação: sequência, iteração, condição, variáveis, execução paralela, sincronia, interação em tempo real, lógica booleana, números randômicos, tratamento de evento e criação de interfaces. A figura 1 mostra o layout da interface do software scratch.

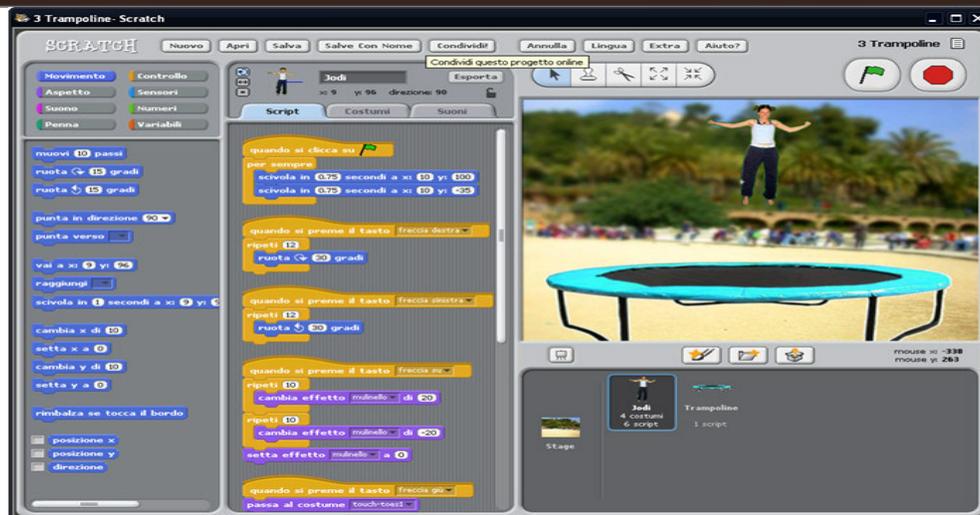
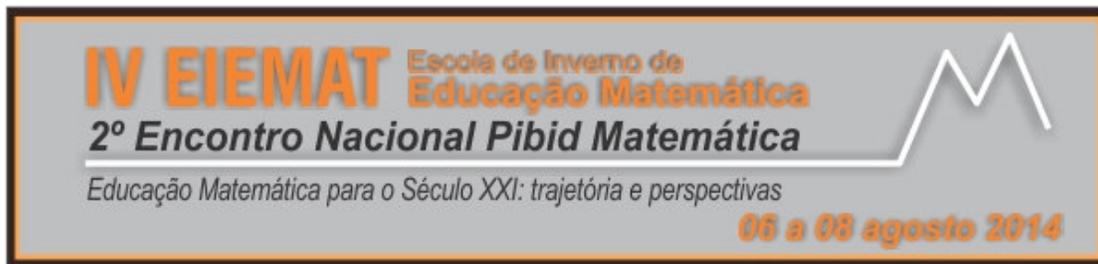


Figura 1 – Interface do software scratch com script de dois objetos e cenário
 Fonte: <<http://scratch.mit.edu/explore/projects/animations>>

Além dos conceitos de programações descritos acima, o software scratch proporciona através de seus comandos “variáveis”, “operadores”, “sensores” e “controle” os recursos necessários para realizar, entre outras possibilidades, operações matemáticas com ou sem substituições de variáveis, construções de figuras geométricas, manipulação das coordenadas cartesianas, raciocínio lógico usando condicionalidades do tipo “se, senão” e movimentos de objetos/scripts.

Pode-se ainda elencar como potencialidades do software, o desenvolvimento da criatividade, a manipulação de mídia, construções de programas que coordenam simultaneamente animações, textos, músicas, sons e gráficos, além de permitir o compartilhamento de suas produções no sítio próprio da web.

Este software possui uma sintaxe mais intuitiva que as outras linguagens de programação como Pascal, Python, Fortran, Cobol e C⁺⁺. Seus blocos de comandos são visíveis e possuem maior diversidade de comandos prontos, representados por blocos, os quais facilitam a produção de estórias multimídias interativas ou qualquer outro tipo de programação. A figura 2 mostra a praticidade da linguagem scratch quando comparada a outras duas linguagens de programação.

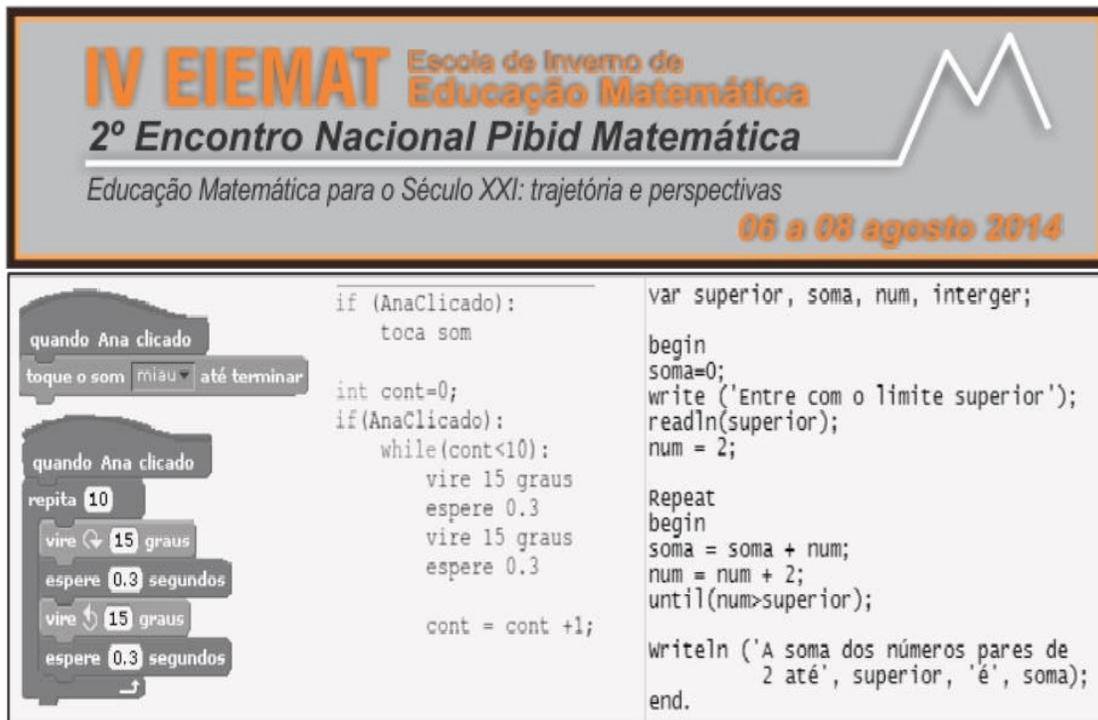


Figura 2 – Apresentação de comando de repetição das linguagens Scratch (esquerda), Python (centro) e Pascal (direita)
 Fonte: SCAICO (2013, p. 96)

O scratch permite a programação através do “arrastamento” de blocos de construção, buildingblocks, que formam pilhas ordenadas, stacks, como se pode perceber nas figuras 3 e 4.

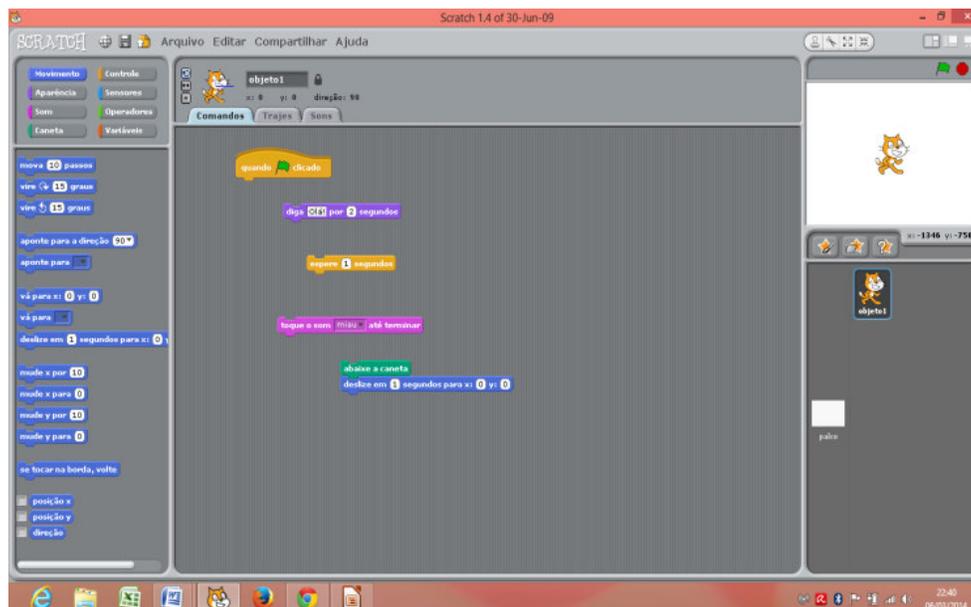


Figura 3 – Comandos (blocos) ainda não encaixados, apenas arrastados para a área de programação, ecrã.³

³Fonte:<https://www.google.com.br/search?q=schatch&source=lnms&tbm=isch&sa=X&ei=TCEZU-DgJtGtAel4oCIBQ&ved=0CAcQ_AUoAQ&biw=1366&bih=606>



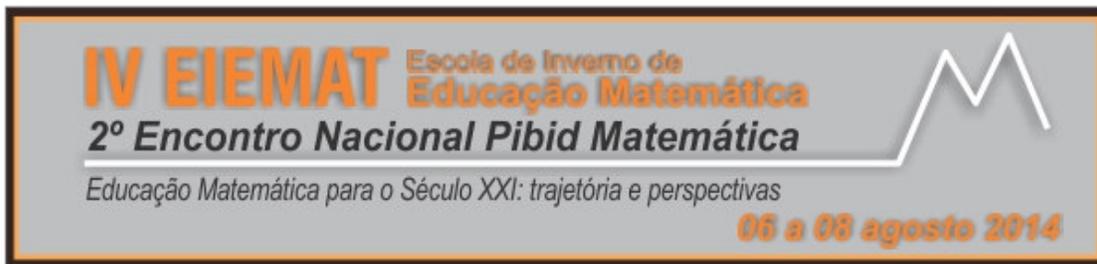
Figura 4 – Comandos (blocos) encaixados formando “pilhas ordenadas”⁴.

Uma das especificidades desta tecnologia é a possibilidade de não se cometer erros de sintaxe, já que não é necessário digitar o comando, pois o mesmo vem em forma de blocos em que basta arrastá-los. Do mesmo modo, os blocos são concebidos para serem encaixados, assim, não possibilitando encaixes com erros. As sequências e as instruções podem ser modificadas mesmo com o programa em andamento, permitindo assim a criação/experimentação de uma nova ideia.

3. A Base Teórica

O minicurso toma como pressupostos teóricos, aqueles onde encontra amparo para desenvolver habilidades de “programação” em ambientes multimídias, portanto com o recurso do computador e dentro de uma prática pedagógica que aborda a construção de conceitos matemáticos. Para isso, uma teoria alicerça este trabalho: a Teoria do Construcionismo de Seymour Papert.

⁴Fonte: <https://www.google.com.br/search?q=scratch&source=lnms&tbm=isch&sa=X&ei=TCEZU-DgJtGtKAE14oCIBQ&ved=0CAcQ_AUoAQ&biw=1366&bih=606>



A teoria do Construcionismo de Seymour Papert dá, nessa dimensão, um dos suportes necessários e preciso sobre as bases desse minicurso.

Seymour Papert nasceu em 1º de março de 1928 na África do Sul. Ele é o teórico mais conhecido sobre o uso de computadores na educação, um dos pioneiros da inteligência artificial e criador da linguagem de programação LOGO, em 1967, que é a base do software de programação scratch, objeto deste estudo.

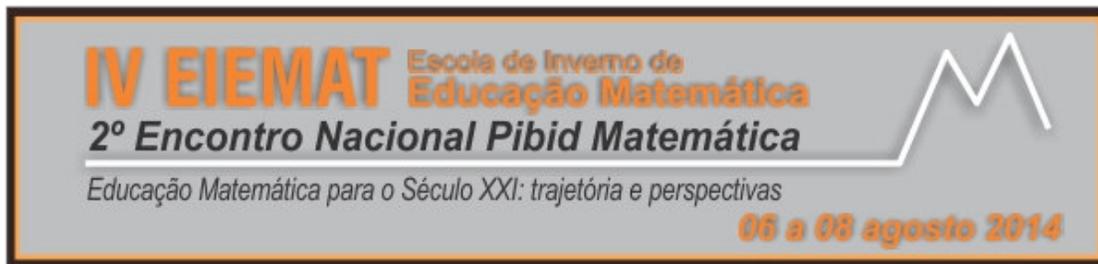
O Construcionismo diz respeito à construção do conhecimento baseada na realização de uma ação concreta que resulta em um produto palpável, desenvolvido com o concurso do computador. Na concepção de Papert (1994), o Construcionismo seria uma extensão do Construtivismo. De uma forma sucinta, esta teoria pode ser caracterizada como um modo de aprendizado que pede a construção de algo para que se possa compreender o seu funcionamento. Implica numa interação aluno-objeto, promovida pelo professor que pode ser organizada através de uma linguagem de programação, como é o caso do scratch.

Papert vai ainda mais longe quando vislumbra no computador um instrumento para a construção dos processos cognitivos das crianças. Ele acredita que o computador permite transpor a barreira do pensamento concreto para o abstrato.

O computador não é somente mais um instrumento educacional poderoso. Ele é o único a nos permitir os meios para abordar o que Piaget e muitos outros identificam como o obstáculo que deve ser transposto para a passagem do pensamento infantil para o pensamento adulto. . . . Conhecimentos que só eram acessíveis através de processos formais podem agora ser abordados concretamente (PAPERT, 1997, p. 37).

A construção de conceitos matemáticos utilizando o computador como recurso e um software de programação como um meio, proporciona uma rica experiência no processo de assimilação do conhecimento e na formação de habilidades dos alunos. A organização simultânea de vários eventos, ordenados de forma independente uns dos outros, proporciona momentos de reflexão, investigação e capacidade de resolução de problemas.

A construção de programas exige do aluno um esforço suplementar de compreensão dos conceitos, muitas vezes obrigando a encará-los sob novas formas, e exige a elaboração de uma estratégia semelhante, em muitos aspectos, às que se usam para enfrentar



situações problemáticas (PONTE, 1991, p.78 apud PINTO, 2010, p. 31).

Nesta concepção, o aluno é visto como autor no processo da aprendizagem, pois exige dele soluções próprias e pertinentes para ultrapassar barreiras e resolver situações problemáticas.

4. Procedimentos Metodológicos: Sequência das Atividades

Para conhecer o software de programação Scratch e verificar as suas possíveis potencialidades no campo do desenvolvimento cognitivo, optou-se por uma sequência de seis atividades iniciais que possibilitarão a compreensão da sua sintaxe.

Na primeira atividade, pretende-se mostrar a interface do programa, explicando os vários campos necessários para as programações.

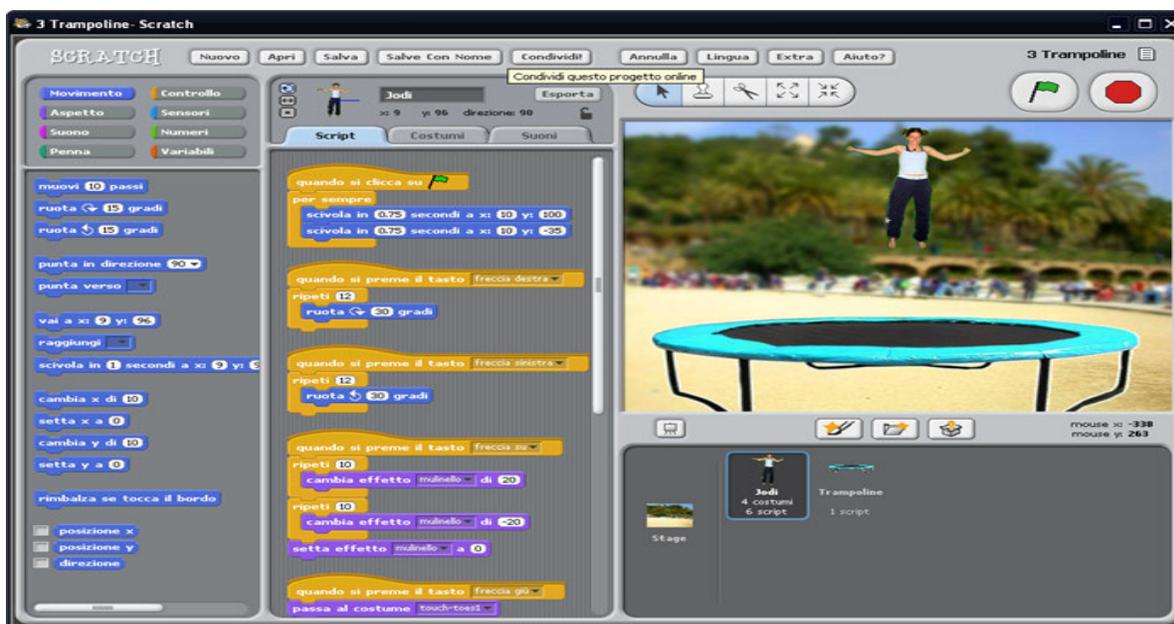


Figura 4 – Interface do software scratch com script de dois objetos e cenário

Fonte: <<http://scratch.mit.edu/explore/projects/animations>>

Na segunda atividade, pretende-se abordar os comandos: “edição de personagem”, “edição de cenário”, “troca de trajes”, “deslizar” e “vire para” através de uma programação que simule um personagem percorrendo uma pista editada em um plano cartesiano como cenário. Com esta atividade, pressupõe-se abordar conceitos matemáticos que envolvam ângulos e coordenadas cartesianas.

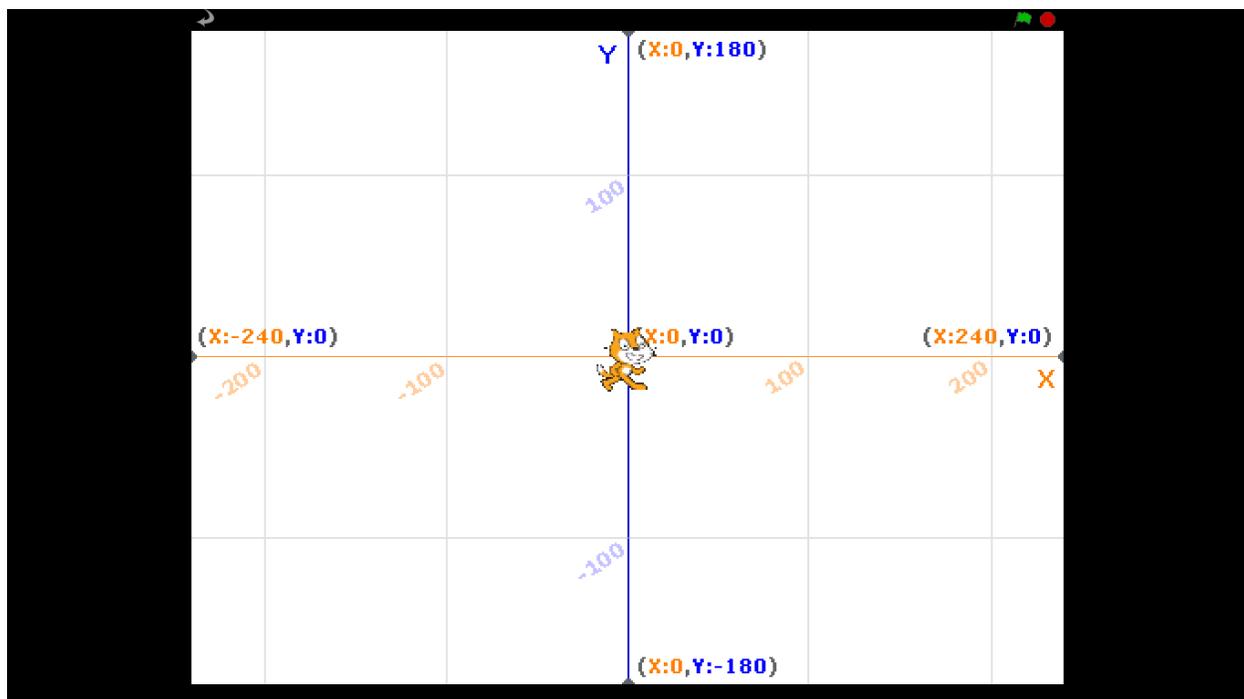


Figura 5 – Imagem do aplicativo

Na terceira atividade, pretende-se abordar os comandos “vire para”, “vá para”, “mude x por”, “mude y por”, “deslizar”, “aponte para a direção” e “caneta” através de uma programação de construção de figuras geométricas como o quadrado, retângulo, triângulo e hexágono. Com esta atividade, busca-se abordar conceitos matemáticos como ângulos, plano cartesiano, polígonos, área e perímetro.

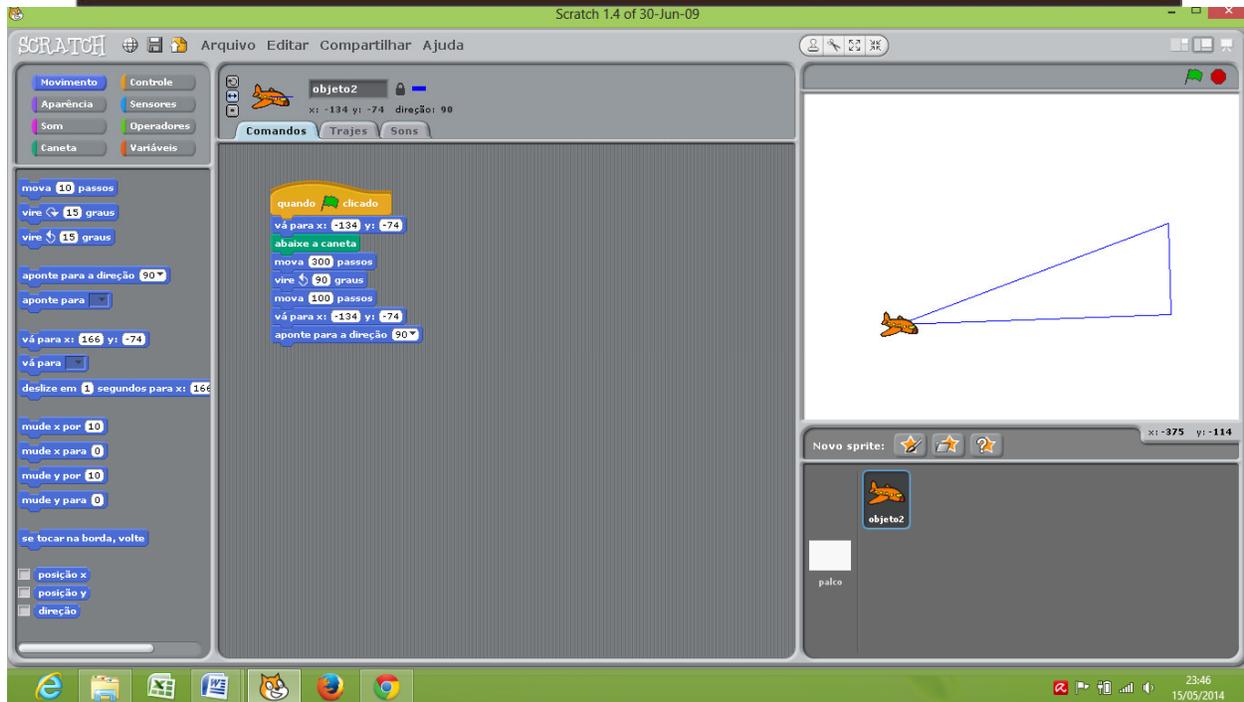
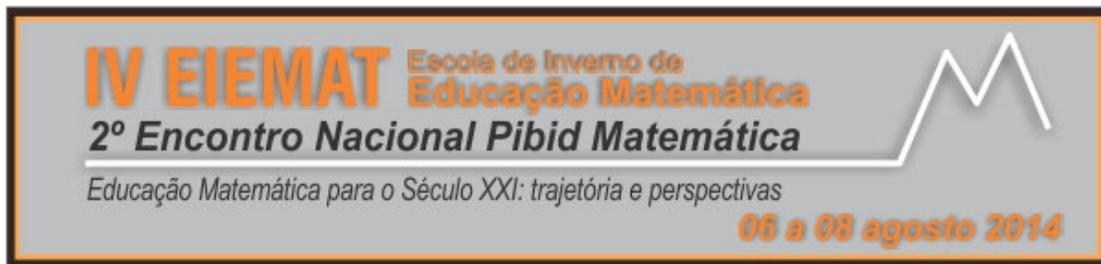


Figura 6 – Imagem do aplicativo

Na quarta atividade, pretende-se utilizar os comandos “controle”, “operadores”, “trajes”, “anuncie para todos e espere” e “movimento” para realizar uma programação de problemas matemáticos envolvendo diálogo entre dois personagens. Com esta atividade pretende-se abordar as operações matemáticas de adição, subtração, multiplicação e divisão.



Figura 7 – Imagen do aplicativo

Na quinta atividade pretende-se utilizar o comando “se senão”. É o chamado “fluxo linear”, porque a ação começa numa ponta (a de cima) e acaba noutra (a de baixo). Só serão cumpridos os comandos contidos na primeira “boca” ou na segunda “boca”, conforme a condição escrita em \diamond seja ou não verdadeira. Com esta atividade pode-se abranger operações matemáticas como soma, subtração, multiplicação, divisão, potenciação, ou qualquer outra operação, além de substituição de variáveis através dos comandos “sensores”, também utilizados nesta atividade.

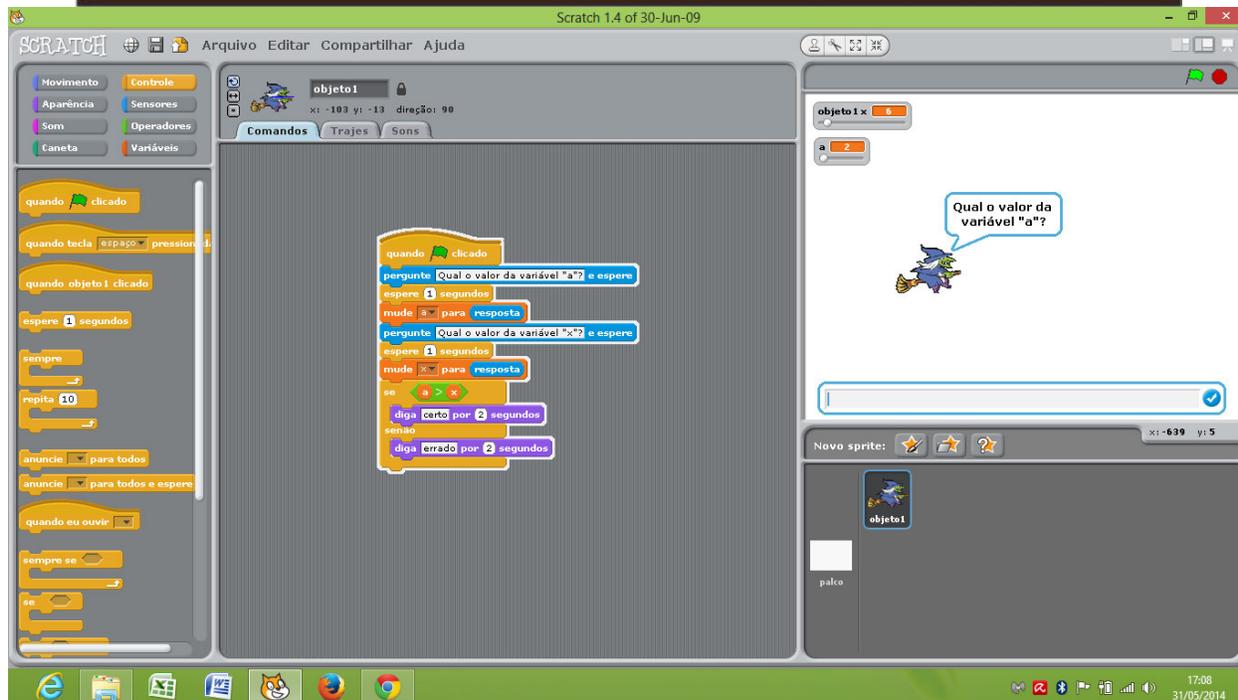
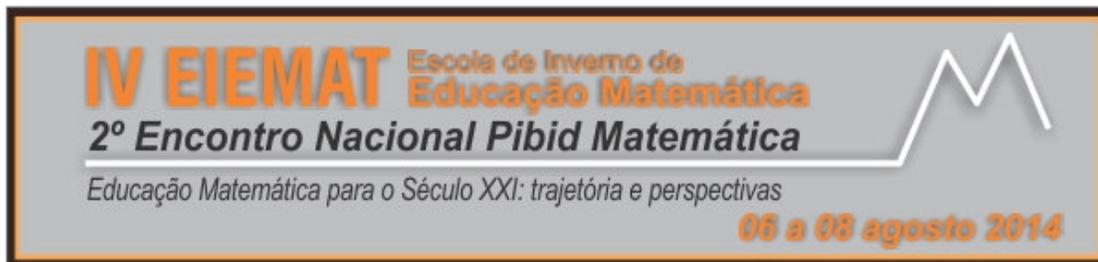
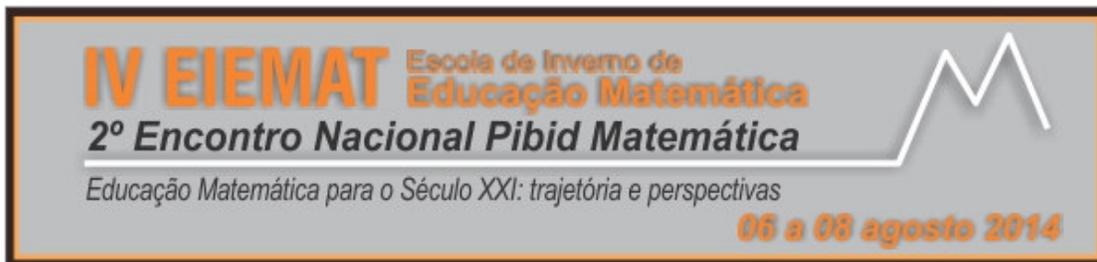


Figura 8 - Imagem do aplicativo

Outras atividades além destas, serão incorporadas ao minicurso para contribuírem na compreensão da programação na linguagem do aplicativo.

Com o desenvolvimento destas atividades, pretende-se utilizar vários comandos de programação, tais como “controle”, “caneta”, “movimento”, “variáveis”, “operadores” e “aparência”, além da edição do cenário e scripts (personagens). As programações acontecem de maneira sincronizada e sequencial, fator fundamental para que os eventos possam acontecer no tempo certo.

Além dos comandos lógicos da programação, percebe-se ainda a utilização de operadores, símbolos e conceitos matemáticos envolvidos nestas construções, como a multiplicação, divisão, coordenadas cartesianas e ângulos. A conexão desses dois procedimentos: comando lógico de programação e das operações matemáticas dão as bases suficientes para as construções desejadas.



Um simples comando para fazer um personagem/objeto mover-se neste ambiente, requer um conhecimento que vai além dos comandos de programação, requer um pensar criativo, lógico, capaz de organizar essas informações.

Com a aquisição da fluência no software Scratch, juntamente com a exploração de atividades desafiadoras, poder-se-á verificar e analisar junto aos usuários a construção de sequencias de comandos que mostram a exploração de estruturas algébricas, na medida em que se incorpora nesta programação variáveis envolvendo números e operações matemáticas; a utilização de elementos da geometria analítica, na medida em que se manipula coordenadas cartesianas ao localizar ou mover personagens e objetos; a utilização de elementos da geometria plana, na medida que se constroem figuras geométricas.

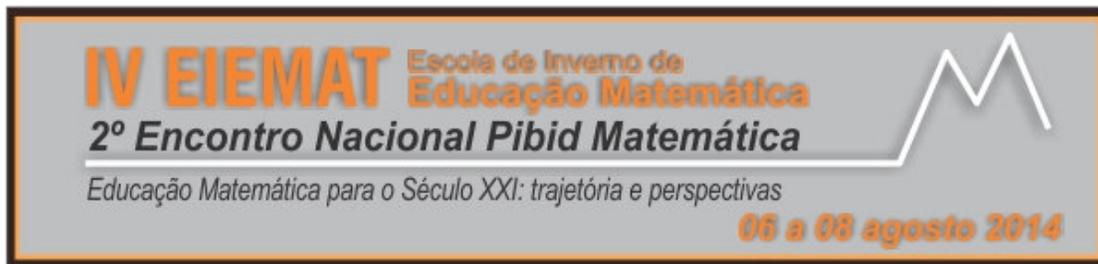
Os resultados esperados ao executar programações que envolvam conceitos matemáticos, pressupõem ainda, uma mudança no processo de construção e assimilação de novos conhecimentos, pois faz com que os alunos, ao interagirem com o computador e a linguagem de programação scratch, desenvolvam habilidades tais como: investigação, raciocínio lógico, antecipação de procedimentos (capacidade de prever situações futuras a partir de comandos anteriores), entre outras, até então pouco exploradas no ensino da matemática e que lhes proporcionarão maior capacidade de interpretação e resolução de situações problemas e desafios.

O software scratch instiga e promove a criatividade, a imaginação e a autonomia dos alunos na medida em que eles próprios constroem as engrenagens e o percurso da sua programação lógica através dos comandos conexos com os conhecimentos e conceitos matemáticos.

5. Considerações Finais

Neste minicurso apresenta-se uma proposta diferente de abordar alguns conceitos matemáticos com a utilização do Software de Programação Scratch.

Os aspectos abordados foram as TIC, o software de programação scratch, o Construcionismo e os procedimentos metodológicos utilizados na pesquisa.



Com o minicurso finalizado, busca-se apresentar as noções básicas desta linguagem de programação, contribuir para o fortalecimento de novas práticas educativas nesta área do conhecimento e contribuir ainda para as pesquisas relacionadas à construção de conceitos matemáticos através de programações com o scratch.

6. Referências Bibliográficas

PAPERT, S. **A família em Rede**. Lisboa: Relógio D'Água, 1997.

PAPERT, S. **A Máquina das Crianças**: Repensando a Escola na Era da Informática. Porto Alegre: Artes Médicas, 1994.

PINTO, A. S. **Scratch na aprendizagem da Matemática no 1.º Ciclo do Ensino Básico**: estudo de caso na resolução de problemas. 2010. 119 f. Dissertação (Mestrado em Área de Especialização em Estudos da Criança Tecnologias de Informação e Comunicação)- Universidade do Minho, Braga, 2010.

SCAICO, P. D. et al. **Ensino de programação no ensino médio**: uma abordagem orientada ao design com a linguagem scratch. Revista Brasileira de Informática na Educação, v. 21, n. 2, p. 92-103, 2013.

SCRATCH, Disponível em: <http://scratch.mit.edu/explore/?date=this_month>. Acesso em 02 mar. 2014.

VALENTE, J. A. **Computadores e conhecimento**: repensando a educação. Campinas: UNICAMP, 1993.