

PRODUÇÃO DE MATERIAL DIDÁTICO PARA O SCRATCH: UMA FERRAMENTA DE APOIO AO ENSINO DE PROGRAMAÇÃO

Maria Gracielly Fernandes Coutinho

IFRN - Nova Cruz.

<http://lattes.cnpq.br/3355046178077879>

<https://orcid.org/0000-0001-8167-5568>

E-mail: gracielly_fernandes23@hotmail.com

Rubiany Farias Mendes

IFRN - Nova Cruz.

<http://lattes.cnpq.br/7863978917511301>

<https://orcid.org/0009-0009-9638-1402>

E-mail: biany.mendes@gmail.com

Jalerson Raposo Ferreira de Lima

IFRN - Nova Cruz.

<http://lattes.cnpq.br/4433845814879129>

<https://orcid.org/0009-0002-7275-236X>

E-mail: jalerson@gmail.com

DOI-Geral: <http://dx.doi.org/10.47538/RA-2024.V3N3>

DOI-Individual: <http://dx.doi.org/10.47538/RA-2024.V3N3-37>

RESUMO: Este trabalho foi desenvolvido com o objetivo de propor a utilização do Scratch como uma ferramenta de apoio no ensino de programação. Uma pesquisa foi realizada a fim de identificar como os alunos avaliavam a disciplina de algoritmos. Os dados coletados revelaram que a grande maioria dos participantes já havia sentido dificuldade na disciplina em questão, porém também foi grande a porcentagem dos que afirmaram gostar desta área. Tendo em vista estes resultados, desenvolveu-se um material didático de forma simples e dinâmica, para ensinar aos interessados como utilizar o aplicativo proposto.

PALAVRAS-CHAVE: Scratch; Programação; Material didático; Ensino.

PRODUCTION OF TEACHING MATERIALS FOR SCRATCH: A TOOL TO SUPPORT PROGRAMMING TEACHING

ABSTRACT: This work was developed with the aim of proposing the use of Scratch as a support tool in teaching programming. A survey was carried out in order to identify how students evaluated the algorithms subject. The data collected revealed that the vast majority of participants had already experienced difficulty in the subject in question, but there was also a large percentage of those who said they liked this area. In view of these results, teaching material was developed in a simple and dynamic way, to teach interested parties how to use the proposed application.

KEYWORDS: Scratch; Schedule; Didactic material; Teaching.

INTRODUÇÃO

A construção de programas de computador exige, entre outras habilidades, a capacidade de resolver problemas através da identificação de um conjunto ordenado e finito de etapas e/ou instruções que levam a sua resolução. Essas etapas são denominadas algoritmo, o qual consiste na descrição de uma sequência de passos que deve ser seguida para a realização de uma tarefa. Em nosso cotidiano, executamos toda e qualquer atividade utilizando algoritmos, mesmo não percebendo isso.

Na programação, algoritmos são muito utilizados para descrever as etapas que precisam ser realizadas para que um determinado programa execute as tarefas que lhe são denominadas. Esse processo exige do indivíduo um raciocínio lógico. A utilização da lógica na vida do indivíduo é constante, visto que é ela quem possibilita a ordenação do pensamento humano.

Desenvolver um algoritmo é desenvolver a resolução de um problema. Tarefa na qual o ato de raciocinar logicamente é primordial para a conclusão de todas as etapas. Segundo Steve Jobs, todas as pessoas do mundo deveriam aprender a programar um computador, porque isto ensina a pensar. A programação de computadores está ganhando um espaço considerável na área da educação, já que a mesma inclui a construção de ferramentas que apoiam o processo de ensino e aprendizagem dos estudantes.

A iniciativa Code.org busca despertar o interesse de jovens e crianças pela programação. A mesma conta com o apoio e a participação de grandes nomes, como Bill Gates, criador da Microsoft, Mark Zuckerberg, fundador do Facebook, Jack Dorsey, criador do Twitter, Drew Houston, criador do Dropbox, entre outros. Os pioneiros dessa organização acreditam que todas as escolas poderiam ensinar as crianças a programarem.

Iniciativas como a do Code.org são necessárias tendo em vista a crescente necessidade de programadores numa sociedade cada vez mais tecnológica. Esse tipo de iniciativa é fundamental, considerando a dificuldade apresentada por estudantes de computação, discutida nesse trabalho.

A estrutura deste trabalho está organizada em nove itens. Inicia-se com a justificativa, que remete as dificuldades apresentadas pelos alunos dos cursos na área de tecnologia. Em seguida, são tratados os objetivos, onde é evidenciada a abordagem

proposta para prover melhoria na qualidade do ensino aprendizagem de programação. No terceiro, é apresentada a pesquisa de opinião sobre as dificuldades no aprendizado da programação a fim de identificar as principais deficiências dos alunos, saber o seu conhecimento sobre o Scratch e também se os mesmos possuíam alguma dificuldade ao cursar a disciplina de algoritmo. No quarto item, o questionário de pesquisa onde são detalhadas as perguntas e respostas. Já no quinto é tratada a análise dos resultados da pesquisa, e a partir do cruzamento de respostas. O sexto item trata da ferramenta Scratch e de como a mesma é composta. Logo após, no sétimo item é relatada a elaboração de material didático. E por fim, os itens oito e nove, mostram as conclusões das autoras e as propostas de trabalhos futuros, respectivamente.

JUSTIFICATIVA

Não é de hoje que muitas pessoas iniciam um curso na área de programação, e desistem ainda nos primeiros semestres. Muitos o fazem por sentir grande dificuldade em aprender e assimilar os conteúdos das disciplinas introdutórias. A disciplina de algoritmos é campeã em provocar medo e insegurança nos alunos, já que a mesma exige o desenvolvimento de estratégias de solução de problemas, além de possuir um dos maiores índices de reprovação nos cursos de tecnologia.

Grande parte dos alunos apresentam dificuldades ao iniciar algum curso na área de programação, fator que ao longo dos anos vem acarretando uma grande preocupação. Tendo em vista esse elemento, constata-se que ensinar programação é um dos sete desafios na educação em Computação.

O ato de ensinar alguém a descrever seu raciocínio lógico por meio de comandos é desafiador ao professor e ao aluno. São identificados três tipos de aspectos que contribuem para a dificuldade de aprendizagem de algoritmos, que são: Problemas de natureza didática, Problemas de natureza cognitiva e Problemas de natureza afetiva.

- Problemas de natureza didática: envolve a diversidade de cultura e experiência dos alunos iniciantes, o excesso de alunos por turma, o que impede que os professores lhes ofereçam atendimento e avaliação individual, a dificuldade de expressar e assimilar

a lógica desenvolvida. Além dos problemas que dizem respeito à escolha equivocada do curso;

- Problemas de natureza cognitiva: aspectos que normalmente estão associados à vivência anterior ao ingresso na universidade. Podem ser identificados em alunos cujos estudos foram interrompidos há algum tempo, ou que o ensino médio não os permitiu desenvolver adequadamente as capacidades cognitivas;

- Problemas de natureza afetiva: estados afetivos que criam um bloqueio na mente, não permitindo que a informação passada pelo professor seja captada de forma eficiente. Goleman (1999) acredita que alunos que apresentam sentimentos de ansiedade, raiva ou depressão têm dificuldade em aprender.

A fim de identificar empiricamente se os alunos regulares dos cursos de informática do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Campus Nova Cruz, também apresentam essas dificuldades discutidas anteriormente, foi aplicado um questionário online para levantamento de dados.

Após análise dos dados coletados, os resultados indicam que 70% dos alunos de informática têm ou já tiveram dificuldade na disciplina em questão. Apesar disso, grande quantidade também afirmou gostar da área de programação cerca de 76%.

Portanto, constata-se que, apesar dos alunos gostarem de programação, é grande a deficiência na aprendizagem dos mesmos. O ensino da disciplina de Algoritmos pode ser considerado como alfabetização da programação de sistemas computacionais. Esse processo de ensino-aprendizagem requer o domínio de operações como pensar, analisar, sintetizar e avaliar.

OBJETIVOS

Devido a constante dificuldade que muitos alunos apresentam ao iniciarem o aprendizado de programação, foi realizada uma pesquisa a fim de identificar as principais deficiências dos alunos. Além disso, procura-se obter resultados sobre os seus conhecimentos em relação o Scratch e também quais as maiores dificuldades apresentadas pelos mesmos na disciplina de algoritmos.

O objetivo da abordagem proposta é prover uma melhoria na qualidade do ensino-aprendizado de programação, a partir de uma ferramenta alternativa chamada Scratch. Esta busca introduzir noções de lógica de programação, a partir de blocos de comandos que são encaixados, para movimentar os personagens e desse modo dar vida ao projeto.

O objetivo deste trabalho é mostrar como o Scratch pode contribuir no ensino de programação. Já que este não exige nenhum conhecimento prévio sobre nenhuma linguagem de programação.

A partir de estudos realizados sobre o Scratch foi desenvolvido um material didático, constituído de onze aulas, vinte e quatro exercícios e um tutorial autoexplicativos, que procura ensinar programação de uma forma simples e intuitiva.

PESQUISA DE OPINIÃO SOBRE AS DIFICULDADES NO APRENDIZADO DA PROGRAMAÇÃO

A pesquisa envolvendo alunos de informática foi feita com o objetivo de obter uma análise do interesse dos mesmos na área da programação, e da porcentagem de alunos que sentiram dificuldade na aprendizagem da disciplina de algoritmos. Além disso, buscava-se constatar se os alunos possuíam conhecimento sobre o Scratch, ferramenta de apoio ao ensino da programação.

O público-alvo desta pesquisa trata-se dos alunos do Curso Técnico em Informática do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Campus Nova Cruz, que já haviam cursado ou estavam cursando a disciplina de algoritmos durante a referida pesquisa.

O questionário foi aplicado por meio do serviço de armazenamento e sincronização de arquivos do Google, conhecido como Google Drive. Os alunos tiveram acesso ao questionário através do site do professor orientador deste trabalho. A divulgação foi feita de forma oral pelas autoras desta pesquisa, e também pelos professores de informática do campus, que divulgaram o questionário em suas respectivas turmas.

QUESTIONÁRIO DE PESQUISA

O questionário foi utilizado como ferramenta de pesquisa avaliativa da disciplina de algoritmos. Além disso, buscava-se também analisar o conhecimento dos alunos sobre o Scratch, ferramenta de apoio à aprendizagem de programação. Um total de 71 alunos regularmente matriculados, participaram da pesquisa.

A primeira pergunta do questionário indagava sobre a modalidade de curso em que o respondente estava matriculado, se integrado ou subsequente. As respostas dessa pergunta indicam que 33% dos alunos eram do integrado, enquanto 77% dos alunos pertenciam ao curso subsequente de informática.

A segunda pergunta solicitava que o aluno especificasse o ano, se fosse aluno do integrado, ou o período, se fosse aluno do subsequente. De acordo com os resultados dos questionários, foi observado que houve participação de alunos de todos os períodos/anos das modalidades de curso integrada e subsequente.

A terceira pergunta buscava identificar se os alunos possuíam conhecimento ou não sobre o Scratch. Os resultados dessa questão apontam que, dentre os alunos que participaram da pesquisa, apenas 10% afirmaram conhecer o Scratch, enquanto 90% afirmaram não possuir conhecimento sobre esta ferramenta.

A quarta pergunta foi destinada apenas aos alunos que afirmaram conhecer o Scratch na pergunta anterior, a mesma indagava sobre o que mais chamou atenção dos alunos nesse programa. A partir das respostas obtidas, foi possível constatar que os fatores mais atraentes aos alunos são: a interface colorida e interativa que estimula o aprendizado, a facilidade que o usuário possui para construir jogos, além de despertar o raciocínio lógico em pessoas de todas as idades.

A quinta pergunta questionava sobre o nível de apreciação dos alunos pela área da programação. Foi possível concluir que apenas 7% dos alunos participantes não gostam de programar, enquanto 17% são indiferentes ou gostam razoavelmente. Além disso, 42% afirmaram gostar um pouco, e 34% gostar muito desta área.

A sexta pergunta buscava obter uma justificativa da resposta anterior. As justificativas apontam que muitos alunos gostam de programar, no entanto sentem dificuldades em assimilar os conteúdos. Há também aqueles que não gostam desta área

pelo mesmo motivo. Alguns afirmam se identificar com a área de programação, pois acreditam que esta é de primordial importância. Outros são atraídos pela possibilidade de desenvolver programas e jogos a partir de códigos.

A sétima pergunta indagava se o aluno possuiu ou possuía alguma dificuldade na disciplina de algoritmos. Foi constatado que 70% dos alunos afirmaram possuir ou ter possuído alguma dificuldade na disciplina de algoritmos, enquanto 30% afirmaram não sentir dificuldade.

A oitava pergunta é destinada apenas aos alunos que responderam “Sim” na pergunta anterior. Nela foi solicitado que o aluno discorresse sobre as dificuldades que teve na disciplina de algoritmos. Com base nas respostas, foi possível constatar que as maiores dificuldades apresentadas pelos alunos participantes, são: a declaração de variáveis, a não compreensão da sintaxe dos comandos de repetição, a não compreensão dos problemas propostos pelo professor. Alguns afirmam ainda entender o assunto na teoria, mas, no entanto, não conseguem colocá-lo em prática.

A nona e última pergunta do questionário solicitava que o aluno descrevesse sugestões para ajudar na aprendizagem da disciplina de algoritmos. As principais sugestões dos alunos foram: ferramentas de apoio que atraíssem e chamasse atenção deixando o conteúdo mais divertido, aplicativos interativos, novas técnicas que facilitem o aprendizado, aulas mais dinâmicas e mais exercícios práticos, tutoriais de apoio, mais aulas de reforço com os professores e mais tutores para auxílio extraclasse.

ANÁLISE DOS RESULTADOS DA PESQUISA

Através dos resultados das perguntas do questionário, é possível realizar algumas análises por meio do cruzamento das respostas, e dessa forma tirar algumas conclusões.

A questão três deixa claro, através dos resultados, que a esmagadora maioria dos alunos desconhece o Scratch (90%). Essa constatação reforça a necessidade de trabalhos como esse, que tem como objetivo a divulgação desse tipo de ferramenta.

A questão cinco apresenta resultados que mostram que a grande maioria alunos tem interesse em programar (76%). Contudo, as respostas da questão sete mostram, em contraponto, que boa parte dos alunos sente dificuldade em algoritmos (70%). É entre o

interesse e a dificuldade apresentada pelos alunos que o Scratch tem muito a oferecer no processo de ensino-aprendizado.

Vale notar também que os alunos, ao responderem a nona pergunta, deixaram claro que os recursos que lhes faltam para aprender são aqueles oferecidos pelo Scratch.

SCRATCH

O Scratch é uma linguagem de programação inspirada nas linguagens Logo e Squeak, desenvolvida por *Lifelong Kindergarten Group no Media Lab do MIT (Massachusetts Institute of Technology)*, com financiamento da National Science Foundation, Intel Foundation, Nokia e do consórcio de pesquisa do MIT Media Lab. Foi criada com o propósito de introduzir a programação e a conceitos matemáticos, ao mesmo tempo em que incentiva o pensamento criativo, o raciocínio sistemático e o trabalho colaborativo.

A versão 1.4 deste aplicativo se encontra disponível para download e pode ser instalado nas plataformas Windows, Mac e Linux. Com ele é possível à criação de histórias animadas, jogos, entre outros programas interativos. Sua sintaxe é comum a muitas linguagens de programação, diferenciando-se apenas por não possuir nenhum tipo de complexidade em sua pontuação.

A aplicação do Scratch em diversos projetos vem contribuindo positivamente para a criatividade, a resolução de problemas e a colaboração de seus usuários. O processo de aprendizagem ocorre a partir de uma interface intuitiva que permite ao usuário adicionar mídias, tais como sons e imagens.

O Scratch é uma excelente ferramenta para o ensino de conceitos de Lógica de Programação por possuir uma interface visual amigável e simples levando, de forma intuitiva, às principais estruturas de uma linguagem como: variáveis, operadores, estruturas de decisão e de repetição, e outros.

Algumas instituições já fazem uso do Scratch como uma abordagem alternativa no processo de ensino-aprendizagem. Foram obtidos resultados positivos na utilização do

Scratch como ferramenta de apoio ao ensino da disciplina introdutória de programação na Universidade de Harvard.

Os projetos do Scratch são compostos por um palco e diversos Sprites. O palco é um plano de fundo onde os Sprites executam determinadas ações. Já os Sprites são objetos que se movimentam no palco a partir de blocos de comandos que são encaixados de modo lógico. Na figura 1 é possível visualizar a interface do Scratch.



Figura 1: Interface do Scratch

- **Blocos de Instruções:** Consiste em uma lista de palhetas, que são divididas em oito grupos, sendo eles: Movimento, Aparência, Som, Caneta, Controle, Sensores, Operadores e Variáveis. Cada qual possui uma cor específica e blocos de comandos distintos.
- **Lista de Instruções:** Onde se encontram os blocos de comandos específicos de cada palheta. Podendo ser visualizados de acordo com a palheta selecionada nos blocos de instruções.
- **Informações dos Sprites:** Área onde os blocos de comandos são arrastados e agrupados, formando a estrutura do projeto.
- **Cenário:** Palco onde os Sprites interagem quando o projeto é executado.
- **Visualização do Jogo:** Local onde é visualizado o projeto em execução.

- Sprites: Objetos que executam as ações definidas a partir dos blocos de comandos.

Os blocos de comandos do Scratch são semelhantes ao quebra-cabeça, pois estes possuem formas, tamanhos diferentes que encaixam apenas em determinados locais. São divididos em três tipos: empilháveis, blocos que possuem encaixes no topo, alguns com áreas onde se podem escrever números; chapéus, blocos que têm um topo arredondado, estes devem ser colocados no topo dos blocos de comandos empilháveis; e repórteres/valores, blocos que são encaixados em certas áreas dos comandos empilháveis.

ELABORAÇÃO DE MATERIAL DIDÁTICO

Tendo em vista a significativa dificuldade dos alunos de cursos de tecnologia nas disciplinas de programação, foi produzido um material didático da ferramenta Scratch, o qual tem a finalidade de auxiliar na aprendizagem da programação, por introduzir conceitos de lógica de forma intuitiva.

O público-alvo deste material didático de apoio à aprendizagem consiste principalmente em alunos de cursos técnicos e superiores da área de tecnologia que estejam cursando as disciplinas introdutórias à programação. O material foi elaborado numa linguagem simples e dinâmica, de modo que qualquer pessoa, seja adulto ou criança, consiga compreendê-lo.

O material consiste em um tutorial, onze aulas explicativas e exemplos de projetos desenvolvidos a partir das aulas. O mesmo foi produzido de acordo com as funcionalidades da ferramenta Scratch. O tutorial funciona como um apoio às aulas. Nele é possível encontrar um detalhamento dos assuntos expostos nas aulas. Todas as aulas, exceto a aula 10, contêm exemplos de projetos desenvolvidos no programa. O mesmo está disponível para download (<http://goo.gl/QUZIYz>).

A primeira aula, denominada aula 0, é destinada a introduzir os conceitos, as funções, e os motivos para utilização da ferramenta. Além disso, é descrito como se deve proceder para fazer o download e a instalação do programa e como a interface do mesmo é composta (http://pt.slideshare.net/Scratch_TCC/aula-0-introduo-ao-scratch-14). Já na aula 1, são descritos os componentes de um projeto, como adicionar e apagar Sprites,

adicionar trajes, escolher palco e como salvar e executar um projeto (http://pt.slideshare.net/Scratch_TCC/aula-1-explorando-o-scratch-14).

As oito aulas seguintes tratam das funcionalidades de cada palheta, as quais são: Movimento, Aparência, Som, Caneta, Controle, Sensores, Operadores e Variáveis, respectivamente. Em todas estas estão contidos além de exemplos, exercícios para praticar, e as funcionalidades dos principais blocos de comandos.

- Aula 2 - Movimento: É trabalhado como movimentar os Sprites dentro do palco, e como alterar a localização dos mesmos no cenário (http://pt.slideshare.net/Scratch_TCC/aula-2-movimento).

- Aula 3 – Aparência: É trabalhado como modificar os trajes dos objetos, alterar cor e tamanho dos mesmos, entre outras funções (http://pt.slideshare.net/Scratch_TCC/aula-3-aparancia).

- Aula 4 – Som: É trabalhado com adicionar sons ao projeto, os quais podem ser gravados ou importados (http://pt.slideshare.net/Scratch_TCC/aula-4-som).

- Aula 5 – Caneta: É trabalhado como traçar linhas no palco enquanto se movimentam os Sprites (http://pt.slideshare.net/Scratch_TCC/aula-5-caneta).

- Aula 6 – Controle: É trabalhado como executar as funções principais do projeto. É nesta palheta que se encontram os blocos de comandos de seleção e repetição (http://pt.slideshare.net/Scratch_TCC/aula-6-controle).

- Aula 7 – Sensores: É trabalhado como utilizar os comandos de entrada e saída de dados. Os blocos de comandos desta palheta são responsáveis por detectar e sinalizar uma condição que se altera ao longo do tempo (http://pt.slideshare.net/Scratch_TCC/aula-7-sensores).

- Aula 8 – Operadores: É trabalhado como calcular novos valores usando variáveis ou constantes. Nesta palheta estão contidos operadores aritméticos, relacionais e lógicos (http://pt.slideshare.net/Scratch_TCC/aula-8-operadores).

- Aula 9 – Variáveis: É trabalhado como criar, modificar e utilizar variáveis nos Scratch. Além de trabalhar a criação de listas (http://pt.slideshare.net/Scratch_TCC/aula-9-variveis).

• Na última aula, denominada aula 10, é feita uma comparação entre as versões 1.4 e 2.0 do Scratch, onde são detalhadas as principais diferenças entre elas (http://pt.slideshare.net/Scratch_TCC/aula-10-scratch-14-x-20).

CONCLUSÕES

Não é de hoje que são criadas propostas com o objetivo de auxiliar o processo de ensino e aprendizagem dos conceitos iniciais da programação de computadores. Neste trabalho foi proposta a utilização do software Scratch como uma abordagem alternativa para o processo de ensino-aprendizagem da programação. Na pesquisa realizada constatou-se que grande parte dos alunos do curso técnico em informática teve dificuldades em aprender a programar, o que reafirma a necessidade de novos métodos de ensino.

Diante da grande necessidade de novas estratégias de aprendizagem, foi elaborado um material didático do aplicativo Scratch, contendo tutorial, aulas e exemplos, a fim de que os professores possam utilizá-lo para ensinar disciplinas introdutórias a programação. Tendo em vista que, o uso dessa ferramenta auxiliado a um planejamento pode ser de grande valia para se obter resultados positivos.

TRABALHOS FUTUROS

Tendo em vista a necessidade da experimentação do material didático produzido através deste trabalho, é essencial uma avaliação do mesmo em trabalhos futuros. Considerando a possibilidade de desenvolver grandes projetos com o Scratch, propomos a continuidade na elaboração do material didático para abordar a construção de algoritmos mais avançados.

REFERÊNCIAS

ANTONIO, Marco; MARQUES, Marcelo; VIEIRA, Marcio; et al. **Algoritmos e lógica de programação**. 2. ed rev. e ampl. São Paulo: Cengage Learning, 2011. 232 p.

ARAÚJO, Rogério. **Lógica de Programação e Estrutura de Dados**. Disponível em: http://rogerioaraujo.files.wordpress.com/2011/11/ebo01_logicaprogramacaoestruturadados_capitulo01.pdf. Acesso em: 10 mar. 2014.

CODE.org. Disponível em: <http://code.org/>. Acesso em: 20 mar. 2014.

CRISTINA, Viviane; CABRAL, Patrícia. **Avaliando o uso do Scratch como abordagem alternativa para o processo de ensino-aprendizagem de programação.**

Disponível em:

http://www.imago.ufpr.br/csbc2012/anais_csbc/eventos/wei/artigos/Avaliando%20o%20Uso%20do%20Scratch%20como%20abordagem%20alternativa%20para%20o%20processo%20de%20ensino-aprendizagem%20de%20programacao.pdf. Acesso em: 12 mar. 2014.

GOOGLE DRIVE. Disponível em: <http://drive.google.com>. Acesso em: 29 mar. 2014.

INÊS, Janice; MARQUES, Júlia; MASCHIO, Elisângela; ROCHA, Anita Maria. **O Desafio de Trabalhar com Alunos Repetentes na Disciplina de Algoritmos e Programação.** Disponível em:

http://www.proativa.virtual.ufc.br/sbie/CD_ROM_COMPLETO/workshops/workshop%202/O%20Desafio%20de%20Trabalhar%20com%20Alunos%20Repetentes%20na.pdf. Acesso em: 12 mar. 2014.

MALAN, D. J.; LEITNER, H. H. (2007). Scratch for budding computer scientists. Proceedings do 38th SIGCSE'07, Kentucky, USA, p. 223–227.

MÉLO, F. E. N., et. al (2011). **Do Scratch ao Arduino: Uma proposta para o ensino introdutório de programação para cursos superiores de tecnologia. XXXIX Congresso Brasileiro de Educação em Engenharia, Blumenau, SC.** Disponível em: www.abenge.org.br/CobengeAnteriores/2011/sexoestec/art1886.pdf. Acesso em: 15 mar. 2014.

PEREIRA, P. S.; et. al(2012). **Análise do Scratch como ferramenta de Auxílio ao Ensino de Programação de Computadores. Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Fortaleza – CE.** Disponível em:

www.abenge.org.br/CobengeAnteriores/2012/artigos/104281.pdf. Acesso em: 05 mar. 2014.

RAABE, A. L. A.; SILVA, J. M. C. (2005) “Um Ambiente para Atendimento as Dificuldades de Aprendizagem de Algoritmos”. In: Workshop de Educação em Computação, 13, 2005. Anais. São Leopoldo: UNISINOS, 2005. 1 CD-ROM.

ZANETTI, Humberto; SOUZA, Ana; ABREU, João; et al. **Uso de robótica e jogos digitais como sistema de apoio ao aprendizado.** Disponível em: <http://www.br-ie.org/pub/index.php/pie/article/download/2345/2100>. Acesso em: 14 mar. 2014.

Submissão: fevereiro de 2024. Aceite: março de 2024. Publicação: setembro de 2024.