



UnB

Instituto de
Ciências Exatas

Departamento de
Ciência da Computação

Planos de Aula:
Aprenda Pensamento Computacional e Scratch

Docente orientadora:
Prof.a Dr.a Maristela Terto de Holanda

Discente:
Vinícius Aguiar Monteiro – 16/0072727

Sumário

1.	Introdução	2
1.1.	Público-alvo	2
1.2.	Objetivos	2
1.3.	Estrutura do Curso.....	2
2.	Planos de aula.....	3
2.1.	AULA 01 - Pensamento Computacional	4
2.2.	AULA 02 - Introdução à Programação e ao Scratch.....	7
2.3.	AULA 03 - Procedimentos e Variáveis no Scratch	10
2.4.	AULA 04 - Estruturas de Decisão e Repetição no Scratch.....	12
2.5.	AULA 05 - Processamento de Strings e Finalização e Apresentação do Projeto	14

1. Introdução

Este documento descreve as aulas que compõem o curso intitulado “**Aprenda Pensamento Computacional e Scratch**” que foi desenvolvido como parte do Trabalho de Conclusão do Curso de Licenciatura em Computação na Universidade de Brasília. O curso propõe o ensino do Pensamento Computacional (PC) (pensar como um cientista da computação) e da Linguagem de Programação Visual (LPV) Scratch, de maneira inclusiva, para os estudantes de escolas públicas do ensino médio do Distrito Federal, possibilitando aos estudantes o desenvolvimento das competências e habilidades do Pensamento Computacional e as pretendidas pela Base Nacional Comum Curricular (BNCC) para o ensino médio.

1.1. Público-alvo

O público-alvo deste curso são estudantes do ensino médio público do Distrito Federal. O único pré-requisito para participação é que o estudante seja alfabetizado. A metodologia aplicada será aulas expositivas e práticas em formato de exercícios dos conteúdos aprendidos.

O curso tem carga horária prevista para 10 horas. Cada aula possui duração de 2 horas a serem ministradas presencialmente em um laboratório de informática.

1.2. Objetivos

O objetivo geral do curso é o ensino dos conceitos do Pensamento Computacional (Coleta de dados, análise de dados, representação de dados, decomposição de problemas, abstração, algoritmo e procedimentos, automação, simulação e paralelismo), utilizando o Scratch, para estudantes do ensino médio público do Distrito Federal.

Os objetivos específicos são:

Ensinar os conceitos do Pensamento Computacional e de seus pilares.

Contextualizar os pilares do Pensamento Computacional à realidade dos estudantes.

Ensinar sobre as diferentes formas de representação de algoritmo (pseudocódigo, fluxograma e Scratch).

Ensinar sobre a utilização da ferramenta Scratch.

Ensinar os conceitos e ferramentas do Scratch.

Estimular o aprendizado de linguagens de programação no ensino médio.

Expandir as competências habilidades em lógica de programação.

Estimular o desenvolvimento criativo utilizando jogos.

1.3. Estrutura do Curso

O curso está dividido em dois módulos. O módulo I é referente aos conceitos de Pensamento Computacional e o módulo II é referente ao Scratch.

2. Planos de aula

Os planos de aula dos cinco encontro será detalhado a seguir:

2.1. AULA 01 - Pensamento Computacional	
Informações do curso:	
Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Pensamento Computacional
Carga horária:	2 horas
Objetivos:	
<p>Apresentar o curso e conhecer os estudantes</p> <p>Apresentar o significado de Pensamento Computacional</p> <p>Descrever os quatro pilares para o pensamento computacional</p> <p>Apresentar Conceitos de fluxograma e exemplos</p> <p>Apresentar Conceitos de pseudocódigo e exemplos</p>	
Conteúdos:	
<ul style="list-style-type: none"> • Introdução do curso e ambientação • Conhecendo os estudantes • Pensamento computacional • Pilares do pensamento computacional • Fluxograma • Pseudocódigo 	

Procedimentos:			
Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Introdução do curso e ambientação	0:20	O professor irá se apresentar, apresentar o projeto e o curso que irá ministrar e seu detalhamento. Pedirá aos estudantes que se apresentem com o Nome, a Idade, a Turma e o que os motivou a participar do curso. Este momento tem o objetivo de aproximar os estudantes e conhecê-los melhor para conduzir as aulas.	Slide "Aula 1 - Pensamento Computacional"
Conhecendo os estudantes	0:05	Os estudantes irão preencher o formulário de caracterização do perfil da turma para subsidiar o conteúdo e atividades das próximas aulas.	Questionário "Perfil dos Estudantes"
Pensamento computacional	0:20	Exposição teórica e apresentação do conceito do Pensamento Computacional.	Slide "Aula 1 - Pensamento Computacional"
Pilares do pensamento computacional	0:25	Exposição teórica e apresentação dos pilares do Pensamento Computacional, além de dar exemplos de como aplicá-los na resolução de problemas. A cada explicação de um dos quatro pilares do Pensamento Computacional, é solicitado aos estudantes que deem outros exemplos dos pilares aplicados à vida.	
Fluxograma	0:25	Exposição teórica e apresentação do conceito de Fluxograma como ferramenta de construção do pilar "Algoritmo" do Pensamento Computacional. São apresentados os símbolos e significados das estruturas que formam os fluxogramas. São demonstrados os fluxogramas com exemplos de atividades do cotidiano. Ao final, é passada uma atividade extraclasse sobre construção de fluxogramas.	
Pseudocódigo	0:25	Exposição teórica e apresentação do conceito de Pseudocódigo como ferramenta mais avançada e estruturada de construção do pilar "Algoritmo" do Pensamento Computacional. É apresentada a estrutura do pseudocódigo. Exemplificar problemas do cotidiano, resolvidos em fluxogramas e com pseudocódigos, trazendo assim uma comparação direta entre as duas formas de resolver um problema. Ao final, é passada uma atividade extraclasse sobre construção de fluxogramas e pseudocódigos.	

Avaliação:

A avaliação ocorrerá durante a apresentação dos exemplos dos estudantes. Deve-se observar não só os estudantes que tenham conhecimentos prévios do assunto, mas também o comportamento dos estudantes, como respeito pelo professor, interesse no conteúdo e interação quando solicitado. É extremamente importante que nesse momento o professor observe o contexto social dos estudantes e seus conhecimentos no uso de tecnologias, para que nas próximas aulas futuras possa citar exemplos de conceitos de programação aplicados às situações conhecidas dos estudantes, estimulando a aprendizagem significativa.

Referências:

AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: <https://doi.org/10.5753/wei.2017.3556>.

Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. <https://www.coursera.org/learn/computational-thinking-problem-solving/>, acesso em 2022-07-24.

Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. <https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf>, acesso em 2021-10-06.

2.2. AULA 02 - Introdução à Programação e ao Scratch

Informações do curso:

Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Scratch
Carga horária:	2 horas

Objetivos:

- Apresentar a programação para resolução de tarefas cotidianas para os alunos, o ambiente de programação Scratch e os diferentes tipos de comandos para criação de cenários
- Apresentar os conceitos de algoritmos, descrição narrativas de algoritmos, efeitos sonoros e visuais disponíveis no Scratch, utilização de camadas e criação de cenas com animações

Conteúdos:

- Revisão do conteúdo da aula anterior
- Introdução à programação
- Criando conta no Scratch
- Explorando o Scratch
- Atividade 10 blocos
- Apresentação do Scratch
- Conceitos de algoritmos
- Criação de figuras, efeitos visuais e sonoros no Scratch
- Discussão sobre o projeto

Procedimentos:

Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Revisão do conteúdo da aula anterior	0:10	O professor fará uma breve exposição oral com uma revisão dos conceitos (Pensamento Computacional e seus pilares, fluxograma e pseudocódigo) passado na aula anterior com o objetivo de fixar o conteúdo e tirar alguma dúvida que tenha surgido.	
Introdução à programação	0:20	Exposição teórica e apresentação do conceito de programação.	Slide "Aula 2 - Introdução ao Scratch"
Criando conta no Scratch	0:10	O professor orientará os estudantes a realizarem o cadastro no Scratch. O professor deverá, antes da aula, preparar uma turma no Scratch com o recurso para educadores e disponibilizar o link da turma para os estudantes realizarem o cadastro no link disponibilizado, assim o professor terá	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/

		controle das atividades e poderá propor atividades para todos.	
Explorando o Scratch	0:10	O professor deixará os estudantes livres para explorar a ferramenta para se familiarizarem e, talvez, até arriscarem a criar algo no Scratch. O objetivo de pedir que eles explorem a ferramenta sem antes ter explicado o que é cada coisa é para que se sintam desafiados a explorar cada botão e funcionalidade.	Site https://scratch.mit.edu/
Atividade 10 blocos	0:10	O professor orientará os estudantes a realizarem a atividade prática proposta (10 blocos), após a exploração do Scratch e disponibilizará uma atividade, dentro da turma, no Scratch para compartilhamento dos estudantes após a finalização. Por ser o primeiro contato com a ferramenta, é esperado que os estudantes tenham dificuldades e apresentem dúvidas de como fazer determinadas ações, porém o professor apenas observará o que farão no Scratch.	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/
Apresentação do Scratch	0:10	Exposição prática e apresentação do site da Linguagem de Programação Visual Scratch, navegando no site e mostrando uma visão geral das funcionalidade e estrutura do Scratch para os estudantes.	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/
Conceitos de algoritmos	0:20	Exposição teórica e apresentação dos conceitos de algoritmo descritivo e pseudocódigo, mostrando exemplos práticos de detalhamento de um problema.	Slide "Aula 2 - Introdução ao Scratch"
Criação de figuras, efeitos visuais e sonoros no Scratch	0:20	Exposição teórica, prática e apresentação de como criar figuras, efeitos visuais e sonoros no Scratch utilizando um dos exemplos dos tutoriais disponíveis no Scratch. O professor orientará os estudantes a realizarem a atividade prática proposta (seguir um dos tutoriais do Scratch).	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/

Discussão sobre o projeto	0:10	Nesse momento os estudantes são organizados em duplas ou trios para discutirem as ideias dos projetos e mediará um debate em grupo para mostrar as possibilidades de desenvolvimento do projeto como jogos, histórias, animações, artes interativas, etc.), além de verificar se já pensaram em algum projeto para desenvolver nas próximas aulas.	Slide "Aula 2 - Introdução ao Scratch" e Site https://scratch.mit.edu/
Avaliação:			
A avaliação será feita pelas dúvidas apresentadas pelos estudantes durante a revisão, além do desempenho durante as atividades práticas propostas. O professor irá percorrer o laboratório e observar o desempenho de cada estudante, solucionando dúvidas e auxiliando nas dificuldades apresentadas.			
Referências:			
AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: https://doi.org/10.5753/wei.2017.3556 .			
Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. https://www.coursera.org/learn/computational-thinking-problem-solving/ , acesso em 2022-07-24.			
Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf , acesso em 2021-10-06.			

2.3. AULA 03 - Procedimentos e Variáveis no Scratch

Informações do curso:

Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Scratch
Carga horária:	2 horas

Objetivos:

- Apresentar e Implementar procedimentos em Scratch e técnicas para construção de comandos
- Apresentar e Implementar tipos de dados no Scratch, criar e manipular dados

Conteúdos:

- Revisão do conteúdo da aula anterior
- Definição do projeto
- Procedimentos
- Variáveis
- Desenvolvimento do projeto

Procedimentos:

Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Revisão do conteúdo da aula anterior	0:10	O professor fará uma breve exposição oral com uma revisão dos conceitos (Scratch, figuras e efeitos visuais e sonoros) passado na aula anterior com o objetivo de fixar o conteúdo e tirar alguma dúvida que tenha surgido.	
Definição do projeto	0:10	As duplas ou trios apresentarão as ideias dos projetos e o professor mediará um debate com os grupos para avaliar a viabilidade e escopo do que será desenvolvido por cada um. Aos grupos que não tiverem uma ideia, serão apresentadas três ideias para escolha e definição do escopo.	
Procedimentos	0:15	Exposição teórica, prática e apresentação de procedimentos, contextualizando com exemplos, e como criá-los no Scratch.	
Variáveis	0:25	Exposição teórica, prática e apresentação de variáveis e seus tipos, além dos identificadores, contextualizando com exemplos, e como criá-las no Scratch. É apresentado uma comparação entre um código Scratch e outro em Python, que fazem a mesma coisa, para que os estudantes vejam a dificuldade de linguagens não visuais, ao mesmo	Slide "Aula 3 - Procedimentos e Variáveis" e Site https://scratch.mit.edu/

		tempo que instiguem a querer aprendê-las. O professor orientará os estudantes a realizarem a atividade prática proposta (criar, usar e manipular dados em pelo menos 4 variáveis, além de fazer operações com elas).	
Desenvolvimento do projeto	1:00	O professor irá auxiliar os estudantes no desenvolvimento do projeto, além de dar suporte e dicas, instigando-os a pensarem nas soluções para os problemas que precisam resolver. Os estudantes serão lembrados dos pilares do Pensamento Computacional com o objetivo de fixar a metodologia de resolução de problemas e, depois aplicar à linguagem que estão utilizando, além de exercitar a colaboração na busca da solução com os demais colegas do grupo.	Site https://scratch.mit.edu/
Avaliação:			
A avaliação será feita pelas dúvidas apresentadas pelos estudantes durante a revisão, além do desempenho durante a atividade prática proposta e desenvolvimento do projeto. O professor irá percorrer o laboratório e observar o desempenho de cada estudante, solucionando dúvidas e auxiliando nas dificuldades apresentadas.			
Referências:			
AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: https://doi.org/10.5753/wei.2017.3556 .			
Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. https://www.coursera.org/learn/computational-thinking-problem-solving/ , acesso em 2022-07-24.			
Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf , acesso em 2021-10-06.			

2.4. AULA 04 - Estruturas de Decisão e Repetição no Scratch

Informações do curso:

Curso:	Aprenda Pensamento Computacional e Scratch
Tema:	Scratch
Carga horária:	2 horas

Objetivos:

- Apresentar e Implementar técnicas para resolução de problemas, uso dos comandos de seleção e fluxo de controle
- Apresentar e Implementar estruturas de repetição para execução de comandos eficientes e variáveis de controle

Conteúdos:

- Revisão do conteúdo da aula anterior
- Operadores relacionais
- Operadores lógicos
- Estruturas condicionais
- Estruturas de repetição
- Desenvolvimento do projeto

Procedimentos:

Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Revisão do conteúdo da aula anterior	0:10	O professor fará uma breve exposição oral com uma revisão dos conceitos (Procedimentos e Variáveis) passado na aula anterior com o objetivo de fixar o conteúdo e tirar alguma dúvida que tenha surgido.	
Operadores relacionais	0:10	Exposição teórica, prática e apresentação de operadores relacionais (Maior que, Maior ou igual, Menor que, Menor ou igual, Igualdade e Diferença), contextualizando com exemplos, e como usá-los no Scratch.	Slide "Aula 4 - Condicionais e Repetição" e Site https://scratch.mit.edu/
Operadores lógicos	0:10	Exposição teórica, prática e apresentação de operadores lógicos ("e" e "ou"), contextualizando com exemplos, e como usá-los no Scratch.	
Estruturas condicionais	0:15	Exposição teórica, prática e apresentação de estruturas condicionais (se então e se então - senão), contextualizando com exemplos, e como usá-los no Scratch. O professor orientará os estudantes a realizarem a atividade prática proposta (criar um programa no Scratch que use pelo menos duas diferentes estruturas condicionais).	

Estruturas de repetição	0:15	Exposição teórica, prática e apresentação de estruturas de repetição (sempre, repita x vezes e repita até que ...), contextualizando com exemplos, e como usá-los no Scratch. O professor orientará os estudantes a realizarem a atividade prática proposta (criar um programa no Scratch que use pelo menos duas diferentes estruturas de repetição).	
Desenvolvimento do projeto	1:00	O professor irá auxiliar os estudantes no desenvolvimento do projeto, além de dar suporte e dicas, instigando-os a pensarem nas soluções para os problemas que precisam resolver. Os estudantes serão lembrados dos pilares do Pensamento Computacional com o objetivo de fixar a metodologia de resolução de problemas e, depois aplicar à linguagem que estão utilizando, além de exercitar a colaboração na busca da solução com os demais colegas do grupo.	Site https://scratch.mit.edu/

Avaliação:

A avaliação será feita pelas dúvidas apresentadas pelos estudantes durante a revisão, além do desempenho durante a atividade prática proposta e desenvolvimento do projeto. O professor irá percorrer o laboratório e observar o desempenho de cada estudante, solucionando dúvidas e auxiliando nas dificuldades apresentadas.

Referências:

AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: <https://doi.org/10.5753/wei.2017.3556>.

Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. <https://www.coursera.org/learn/computational-thinking-problem-solving/>, acesso em 2022-07-24.

Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. <https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf>, acesso em 2021-10-06.

2.5. AULA 05 - Processamento de Strings e Finalização e Apresentação do Projeto			
Informações do curso:			
Curso:	Aprenda Pensamento Computacional e Scratch		
Tema:	Scratch		
Carga horária:	2 horas		
Objetivos:			
<ul style="list-style-type: none">• Apresentar e Implementar técnicas para processamento de strings• Finalizar o projeto e apresentar			
Conteúdos:			
<ul style="list-style-type: none">• Revisão do conteúdo da aula anterior• Processamento de strings• Desenvolvimento do projeto• Apresentação do projeto• Avaliação final			
Procedimentos:			
Conteúdo	Tempo previsto	Detalhamento	Material utilizado
Revisão do conteúdo da aula anterior	0:10	O professor fará uma breve exposição oral com uma revisão dos conceitos (Operadores relacionais e lógicos e Estruturas condicionais e de repetição) passado na aula anterior com o objetivo de fixar o conteúdo e tirar alguma dúvida que tenha surgido.	
Processamento de strings	0:10	Exposição teórica, prática e apresentação de processamento de strings, contextualizando com exemplos, e como usá-los no Scratch. O professor orientará os estudantes a realizarem a atividade prática proposta (criar um programa no Scratch que use processamento de strings).	Slide "Aula 4 - Condicionais e Repetição" e Site https://scratch.mit.edu/
Desenvolvimento do projeto	1:00	O professor irá auxiliar os estudantes no desenvolvimento do projeto, além de dar suporte e dicas, instigando-os a pensarem nas soluções para os problemas que precisam resolver. Os estudantes serão lembrados dos pilares do Pensamento Computacional com o objetivo de fixar a metodologia de resolução de problemas e, depois aplicar à linguagem que estão utilizando, além de exercitar a	Site https://scratch.mit.edu/

		colaboração na busca da solução com os demais colegas do grupo. Como é a última aula, os estudantes farão ajustes necessários e corrigirão possíveis erros que ainda não conseguiram resolver para o pleno funcionamento do projeto.	
Apresentação do projeto	0:30	O professor irá chamar cada grupo para apresentar o projeto que fizeram para o curso. Os estudantes deverão apresentar o os integrantes do grupo, o nome e o que o programa faz, usar o programa e por fim abrir o código para mostrar o que cada parte do algoritmo faz).	Site https://scratch.mit.edu/
Avaliação final	0:10	Os estudantes irão preencher o formulário de avaliação do curso. Este questionário tem o objetivo avaliar a percepção dos estudantes sobre diversos itens do curso.	Questionário "Avaliação final do curso"
Avaliação:			
A avaliação será feita pelas dúvidas apresentadas pelos estudantes durante a revisão, além do desempenho durante a atividade prática proposta e desenvolvimento do projeto. O professor irá percorrer o laboratório e observar o desempenho de cada estudante, solucionando dúvidas e auxiliando nas dificuldades apresentadas. A avaliação final do curso será realizada com a entrega e análise da apresentação dos projetos.			
Referências:			
<p>AONO, Alexandre Hild; RODY, Hugo Vianna Silva; MUSA, Daniela Leal; PEREIRA, Vanessa Andrade; ALMEIDA, Jurandy. A Utilização do Scratch como Ferramenta no Ensino de Pensamento Computacional para Crianças. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 25. , 2017, São Paulo. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2017 . ISSN 2595-6175. DOI: https://doi.org/10.5753/wei.2017.3556.</p> <p>Davidson, Susan e Chris Murphy: Computational thinking for problem solving. University of Pennsylvania, 2022. https://www.coursera.org/learn/computational-thinking-problem-solving/, acesso em 2022-07-24.</p> <p>Lab, Creative Computing: Creative computing curriculum. Desenvolvido por Laboratório de Computação Criativa da Escola de Graduação em Educação de Harvard e Traduzido por Rede Brasileira de Aprendizagem Criativa (aprendizagemcriativa.org) e Universidade Estadual de Feira de Santana (UEFS), Brasil, 2019. https://lcl.media.mit.edu/resources/readings/creative-computing-guide.pt.pdf, acesso em 2021-10-06.</p>			