

ALMANAQUE PARA POPULARIZAÇÃO DE CIÊNCIA DA COMPUTAÇÃO

SÉRIE

7

Pensamento
Computacional



Volume 6

DESAFIOS DE PROGRAMAÇÃO CRIATIVA E PENSAMENTO COMPUTACIONAL: DO CONTO AO CÓDIGO COM SCRATCH E VIBOT



Margarida Romero
Maria Augusta Silveira Netto Nunes
José Humberto dos Santos Junior
Luís Antônio dos Santos Silva
Aurélie Roy
Alexandre Lepage

REITOR

Prof. Dr. Angelo Roberto Antonioli

VICE-REITOR

Prof. Dra. Iara Campelo

CAPA, ILUSTRAÇÕES E EDITORAÇÃO ELETRÔNICA

José Humberto dos Santos Júnior

REVISÃO GERAL

Maria Augusta Silveira Netto Nunes

Os personagens e as situações desta obra são reais apenas no universo da ficção; não se referem a pessoas e fatos concretos, e não emitem opinião sobre eles.

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL
UNIVERSIDADE FEDERAL DE SERGIPE

D441d

Desafios de programação criativa e pensamento computacional : do conto ao código com scratch e vibot [recurso eletrônico] / Margarida Romero ... [et al.]. – Porto Alegre : SBC, 2019.

72 p. : il. – (Almanaque para popularização de ciência da computação. Série 7, Pensamento computacional ; v. 6)

ISBN 978-85-7669-482-3

1. Linguagem de programação (Computadores). 2. Scratch (Linguagem de programação de computadores). 3. Computação. I. Romero, Margarida. II. Série. III. Universidade Federal de Sergipe.

CDU 004.438(059)



Margarida Romero
Maria Augusta Silveira Netto Nunes
José Humberto dos Santos Junior
Luís Antônio dos Santos Silva
Aurélie Roy
Alexandre Lepage

ALMANAQUE PARA POPULARIZAÇÃO DE CIÊNCIA DA COMPUTAÇÃO

Série 7: Pensamento Computacional

**Volume 6: Desafios de programação criativa e
Pensamento Computacional: do conto ao código com
Scratch e Vibot**

Porto Alegre/RS
Sociedade Brasileira de Computação
2019

Apresentação

Essa cartilha/gibi foi viabilizado por meio do projeto de Bolsa de Produtividade CNPq–DTII nº306576/2016-3, coordenado pela Prof^a. Maria Augusta S. N. Nunes em desenvolvimento no Departamento de Computação (DCOMP)/Programa de Pós-graduação em Ciência da Computação (PROCC) – UFS. Os públicos alvos dos gibis são jovens e crianças. O objetivo geral deles é fomentar o interesse pela área de Ciência da Computação.

Esse gibi, Volume 6 da Série 7, é uma reprodução adaptada do livro guia de desafios “Défis de programmation créative: du conte au code avec Scratch et Vibot” produzido por Margarida Romero em 2016 na Université Laval no Quebec (ver Bibliografia). Neste sexto gibi da Série do Almanaque para Popularização de Ciência da Computação apresentamos vários desafios de programação criativa e desenvolvimento do pensamento computacional em forma de passatempos que podem ser realizados com a utilização do software de programação visual Scratch de forma plugada ou desenvolvendo alternativas criativas de forma desplugada.

(os Autores)

DESAFIOS DE PROGRAMAÇÃO CRIATIVA E PENSAMENTO COMPUTACIONAL:

Do conto ao código com Scratch e Vibot.

Este sexto gibi da Série do Almanaque para Popularização de Ciência da Computação apresenta vários desafios de programação criativa e desenvolvimento do pensamento computacional que podem ser realizados com o software de programação visual Scratch. Antes de apresentar os desafios, apresentamos a programação criativa o pensamento computacional e os recursos que podem ajudá-lo a se familiarizar-se com o Scratch.

INTRODUÇÃO À PROGRAMAÇÃO CRIATIVA E AO PENSAMENTO COMPUTACIONAL

A **programação criativa** visa engajar os participantes, antes de mais nada, numa abordagem crítica, empática e criativa de **resolução de problemas de uma certa complexidade e autenticidade**, para, em seguida, os tornar responsáveis pela **criação de uma solução técnico-criativa** que usa **estratégias e processos de ciência da computação** para a criação de uma ou mais soluções. A programação criativa via **pensamento computacional** promove um conjunto de estratégias cognitivas e metacognitivas ligadas a modelagem de conhecimento e de processos, à abstração, ao algoritmo e a identificação, decomposição e organização de estruturas complexas para a **solução criativa de problemas** que utilizem **estratégias e processos da vinculados à ciência da computação** (Romero, 2016).



Como parte das competências-chave para o século 21 (#5c21)*, sendo: (1) pensamento crítico (CrT); (2) colaboração (C); (3) criatividade (CR), (4) resolução de problemas (PS) e (5) pensamento computacional (CT). Na figura acima, o pensamento computacional está relacionado à criatividade e à resolução de problemas.

#VibotORobo



Antes de descobrir os desafios da programação criativa e pensamento computacional, convidamos você a ler o conto introdutório sobre programação “Vibot, le robot”, ou na versão na versão

adaptada, Betabot - a representante brasileira da Liga dos Bots para o desenvolvimento do pensamento computacional no Brasil (em português disponível digital). Ou o original em francês/inglês Vibot le robot (disponível digital ou impresso).

O livro original Vibot le robot foi produzido para iniciar crianças, jovens e adultos (de 7 a 77 anos) em programação. A programação é a língua dos robôs, dos computadores e de outras tecnologias. Nós brasileiros criamos a nova personagem, a robô Betabot, que é a representante brasileira da Liga. O Vibot é o representante canadense/francês da Liga, ele é o primeiro personagem da Liga dos bots, ele nasceu em 2016, é considerado o chefe da Liga. A Liga é responsável por desenvolver o pensamento computacional no mundo. A Betabot, que nasceu em 2019, é a representante brasileira da Liga, ela é uma entidade virtual que pode acompanhar as crianças em diversas plataformas, seja em robôs físicos, celulares, relógios tecnológicos/smart, wearables, tablets, computadores etc. Ela age de acordo com sua missão na Liga do Pensamento Computacional (L.P.C.) e, também, age baseada em seu aprendizado constante auxiliando os aprendizes e demais usuários por meio de diversas interfaces. Atuando na região definida para ela dentro da Liga (L.P.C.).

*Você pode ler também os gibis 1, 2 e 3 e 4 da Série 7 que falam sobre pensamento computacional e Scratch, bem como o Gibi 7 da Série 1 que fala um pouco sobre a linguagem Scratch.



BETABOT

Betabot - a representante brasileira da Liga dos Bots para o desenvolvimento do pensamento computacional no Brasil é um gibi para crianças de 7 a 77 anos que introduz programação e robótica.

Os personagens do gibi recebem uma robô como presente. Eles terão que aprender a programá-la para poder brincar com ela. Felizmente, eles podem contar com a ajuda da Rafaela, que é uma especialista em programação e, também, é fera em matemática e ainda mantém um canal no YouTube que se chama



"Fala aí Geek", com dicas de pensamento computacional, programação e matemática (Série 7, Volume 1 do Almanaque para popularização de Ciência da Computação). A história da Betabot apresenta textos em português e seu correspondente em blocos de programação Scratch.

O SOFTWARE DE PROGRAMAÇÃO VISUAL SCRATCH

O Scratch é um software gratuito projetado para desenvolver projetos criativos com blocos de programação. Os blocos podem ser organizados como um quebra-cabeça no espaço de programação e os personagens e fundos de tela podem ser modificados para desenvolver animações e até mesmo mini-jogos como o Mario Bros.

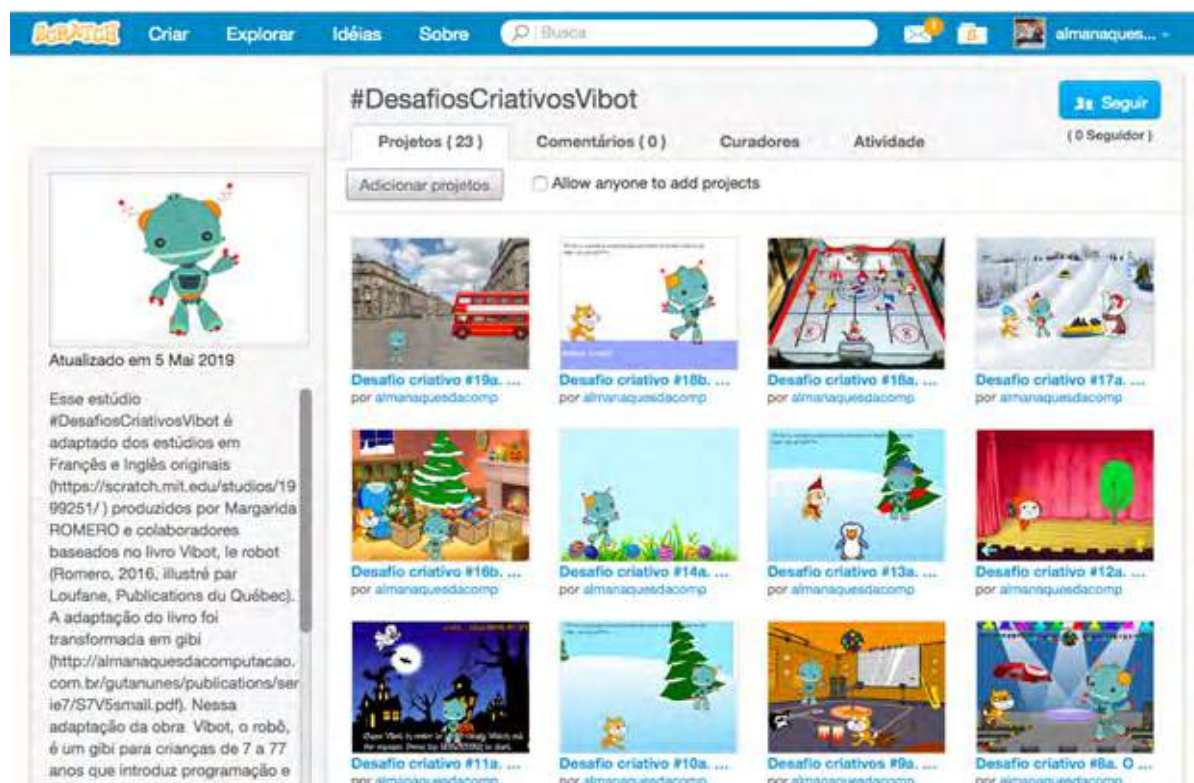
O site <http://scratch.mit.edu> permite consultar milhares de projetos existentes e até remixá-los. Ele também permite que você crie contas do tipo professor ou participante para publicar e compartilhar projetos na Web.



Para se familiarizar com Scratch e a programação criativa, convidamos você a ler o guia [programmation créative avec Scratch de Brennan, Balch et Chung \(2014\)](#) - em francês ou [Creative Computing Brennan, Balch et Chung \(2014\)](#) - em inglês (Não existe tradução em português ainda <http://creativecomputing.gse.harvard.edu/guide/download.html>). Você também encontrará tutoriais em vídeo no Scratch no seguinte endereço: [Video-tutoriais Scratch](#) (vídeos em francês).

ESTUDIO #DESAFIOSCRIATIVOSVIBOT, EXEMPLOS GRATUITOS ONLINE

O estudio #DesafiosCriativosVibot apresenta alguns recursos relacionados aos desafios de programação criativa Scratch. Está disponível em português em: <https://scratch.mit.edu/studios/8379890/> e em inglês/francês em: <https://scratch.mit.edu/studios/1999251/>.



INTRODUÇÃO AOS DESAFIOS DA PROGRAMAÇÃO CRIATIVA

Os desafios da programação criativa e pensamento computacional apresentam uma breve história colocando o gato [Scratch](#) e [Vibot](#) (a Betabot ainda não aparece pois acabou de nascer, mas logo teremos um conjunto de desafios com a representante brasileira da Liga dos Bot - Liga do Pensamento Computacional (L.P.C.)).

Para cada desafio, convidamos você a analisar a história, identificar os componentes (personagens, cenários, ações, ...) e depois programá-los no Scratch, remixando a tela que é proposta para cada desafio criativo.

Antes de começar, nós convidamos você a criar uma conta do Scratch no site <http://scratch.mit.edu>

OS DESAFIOS DA PROGRAMAÇÃO CRIATIVA: DA HORA DO CONTO À HORA DO CÓDIGO

Os desafios da programação criativa que propomos neste almanaque permitem que você desenvolva os componentes do pensamento computacional: de análise, de alfabetização digital e de programação da solução. A figura a seguir mostra os 6 componentes do programação criativa/pensamento computacional (Romero; Lepage; Lille, 2017):

1. Leitura, compreensão e análise de uma história, conto ou situação narrativa (CTc1).
2. A modelagem da situação: a criação de um diagrama ou um roteiro das diferentes cenas do conto, seus personagens e ações (CTc2).
3. Alfabetização para codificar: capacidade de identificar a estrutura do programa e os blocos de código necessários para codificar/construir o algoritmo da história/conto (CTc3).
4. Alfabetização tecnológica e de sistemas: criação de uma conta de usuário no Scratch e gerenciamento de portfólio de projetos (CTc4).
5. A capacidade de criar um programa de computador a partir da análise e modelagem da situação (CTc5).
6. A capacidade de testar e melhorar o programa de computador em uma abordagem reflexiva, ágil e interativa (CTc6).

Da hora do conto à hora do código. Durante essa atividade, os alunos criam um conto ou escolhem um conto existente, depois analisam, modelam e programam.

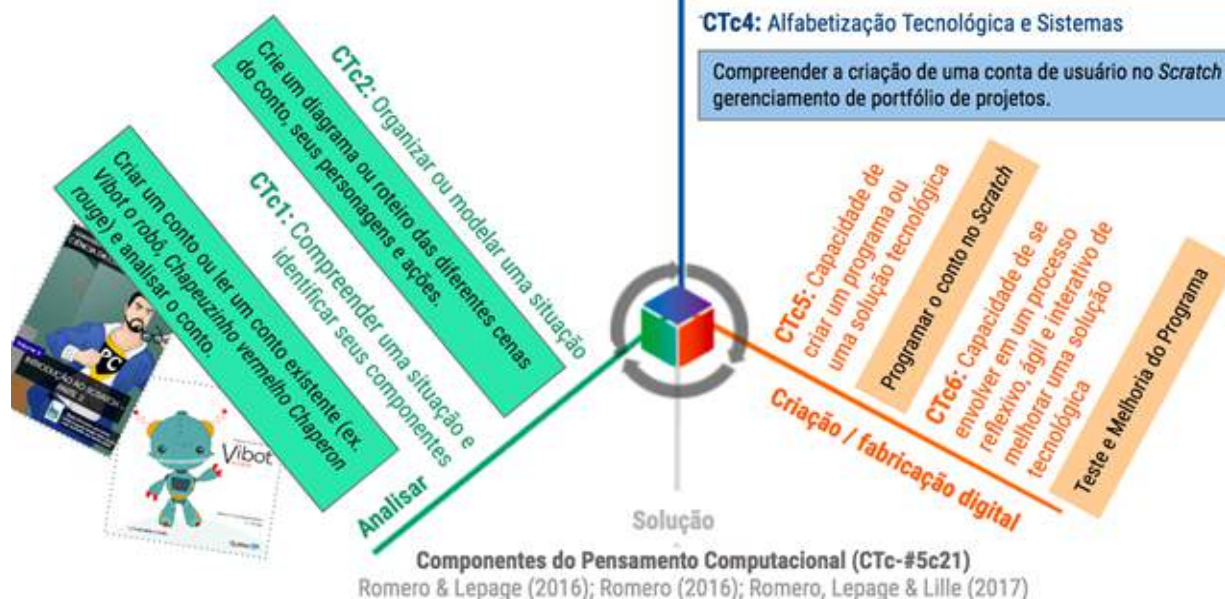
Afabetização digital

CTc3: Capacidade de compreender e desenvolver a lógica de um algoritmo (alfabetização de como codificar)

Identificar a estrutura e os blocos de código necessários para codificar o conto.

CTc4: Alfabetização Tecnológica e Sistemas

Compreender a criação de uma conta de usuário no Scratch e o gerenciamento de portfólio de projetos.



Bom, depois de toda essa introdução, é hora de começar! Olá, nós somos os defensores do Pensamento Computacional e vamos te ajudar a enfrentar todos os desafios que virão a seguir!

Ah! Meu nome é Cícero, e estes são Malu, Ana e Bill.





Desafio criativo #1a. O Vibot e o gato Scratch se apresentam!

Avaliação : Pensamento computacional

- CTc3 (Alfabetização relacionada ao código e algoritmos)
- CTc6 (Abordagem ágil e interativa)

Neste primeiro desafio, convidamos você a corrigir um código textual que entrou nos blocos de programação do Scratch.



Você está pronto para o desafio?

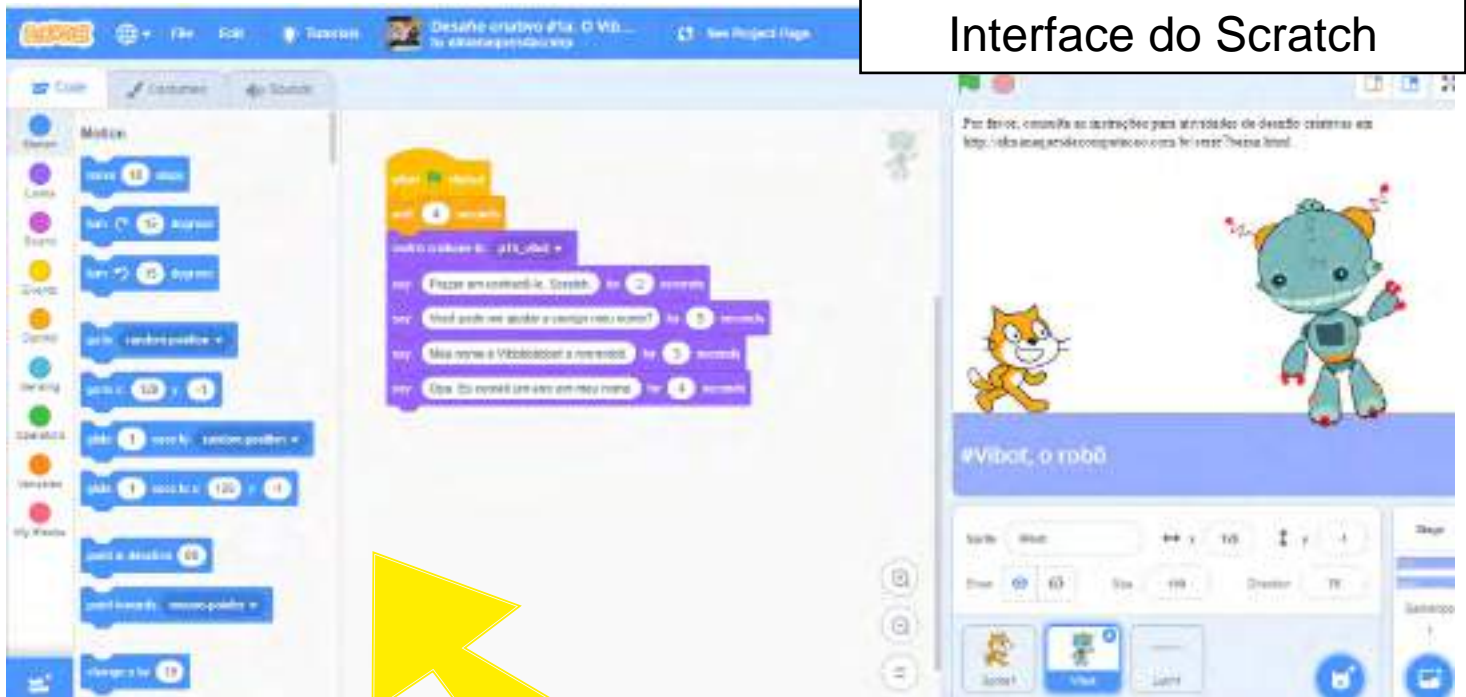
Convidamos você a se conectar ao Scratch para participar do seguinte projeto:

Você pode acessar o projeto nas seguintes linguagens: Francês, Inglês e Português. O Bill te dará a localização dos links.



FR: <https://scratch.mit.edu/projects/144235341/>
EN: <https://scratch.mit.edu/projects/144228583/>
PT: <https://scratch.mit.edu/projects/306785156/>

Interface do Scratch



Na parte esquerda da interface do Scratch, você encontrará os blocos de código correspondentes à animação. Convidamos você a modificar o texto ("Vibbbbbbbbot o rrrrrrrrobô") para que o Vibot possa se apresentar com seu nome verdadeiro ("Vibot o robô").

O texto estará assim:

- Gato Scratch: "Olá!"
- Gato Scratch: "Meu nome é Scratch! E o seu?"
- Vibot: "Prazer em conhecê-lo, Scratch. "
- Vibot: "Meu nome é Vibbbbbbbbot o rrrrrrrrobô."
- Vibot: "Opa. Eu cometi um erro em meu nome. "
- Vibot: "Você pode me ajudar a corrigir meu nome?"

Para realizar o desafio, convidamos você a seguir as duas etapas a seguir:



Etapa 1. Análise da Situação (CTc3).
Onde está o bloco de código que devemos editar?

Etapa 2. Corrija e teste a solução.
No editor Scratch, nós te convidamos a identificar o bloco de código onde o nome do robô Vibot está mal escrito, a fim de corrigi-lo.



Desafio criativo #1b. O Vibot e o gato Scratch se encontram

Convidamos você a criar seu primeiro conto no Scratch a partir de uma tela disponível no seguinte endereço:



FR: <https://scratch.mit.edu/projects/119727640/>
PT: <https://scratch.mit.edu/projects/306819445/>

Sua criação deve corresponder ao conto/história abaixo:



Vibot e Gato Scratch estão no pátio da escola.
Scratch se aproxima do Vibot.

- Gato Scratch: "Oi! Meu nome é Scratch, você é uma criança? "
 - Vibot: "Olá, meu nome é Vibot, eu sou um robô."
 - Gato Scratch: "Eu sei me comunicar com robôs, eu gosto de dar ordens e fazê-los fazer coisas estranhas."
 - Vibot: "Nós robôs podemos executar certas instruções."
 - Gato Scratch: "Eu sei algumas instruções que nos permitirão fazer essas coisas estranhas juntos. Nós nos daremos bem ;-)." "
 - Vibot: "Você pode me dar uma instrução."
 - Gato Scratch: "Desenhe uma linha de cor azul."
- Depois de traçar uma linha azul, o Vibot responde ao Scratch.
- Vibot: "Desenhei uma linha azul."
 - Gato Scratch: "Eu gosto de robôs que obedecem, eu te amo Vibot!"

Antes de começar a programar, convidamos você a ler o conto com atenção e refletir para responder às seguintes perguntas:



Etapas 1. Análise da situação.

- Quais são os personagens?
- Quais são as ações realizadas por cada personagem?
- Onde estão os personagens?

Etapas 2. Modelagem da situação.

Convidamos você a criar um diagrama ou roteiro sobre o desdobramento do conto. Você pode fazer o diagrama em papel ou da forma que preferir.

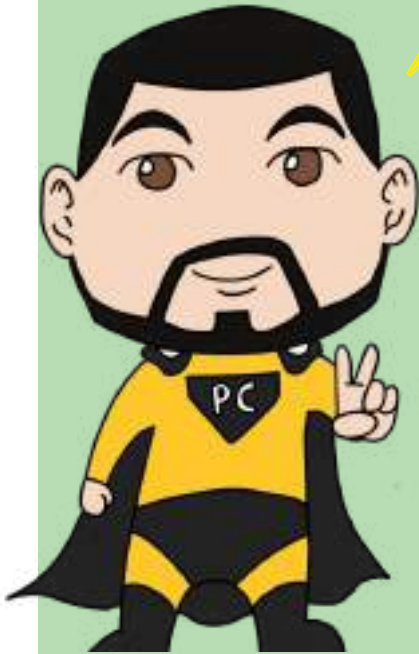
Passo 3. Intuitivamente, escreva os algoritmos (por exemplo, quando o programa inicia, o personagem Vibot se move 60 pixels para a direita).

Convidamos você a criar um diagrama/roteiro antes de realizar sua programação.



Quando o conto/história estiver finalizada, você poderá salvá-la e compartilhá-la.

Desafio criativo #2a. O Vibot colhe uma fruta : modelagem



Avaliação : Pensamento computacional

CTc1 (Análise)

CTc2 (Abstração et modelagem)

Link para as ambiente criativo:

EN : <https://scratch.mit.edu/projects/130395581/>

PT: <https://scratch.mit.edu/projects/306820905/>



(* Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

Nesta história, O Vibot colhe uma fruta. Você não tem mais informações sobre a situação e o cenário, mas mesmo assim, você tem que representá-lo para que ela possa ser executada no Scratch. Nesta fase, no entanto, não se limite sob o pretexto de que você não tem certeza de como fazê-lo no Scratch.



Tarefa esperada: você deve identificar todos os objetos que seriam úteis para representar esta situação e identificar os links que os unem. Você é livre para escolher a forma que mais lhe convier. Seja criativo, desde que a representação da situação seja consistente (não podemos colher frutas em um carro em movimento, por exemplo!).

Você precisa fazer tudo isso no Google Slides, criando um novo documento:
<https://docs.google.com/presentation/u/0/>
Você deve tornar este documento público para que possamos acessá-lo.



Se você não tiver uma Conta do Google, deverá criar uma:

<https://accounts.google.com/SignUp?hl=fr>

Você pode clicar em "Eu prefiro usar meu endereço de e-mail atual" para economizar tempo. Se você está tendo problemas peça ajuda a um colega.

Desafio criativo #2b. O Vibot colhe uma fruta : criação



Avaliação : Pensamento computacional

CTc3 (Algoritmos)

CTc5 (Programação)

Agora você é solicitado a criar no Scratch o conto/história que você modelou (Desafio #2a). Nesta etapa, você pode fazer adaptações ao que você modelou anteriormente, por exemplo, pode usar imagens da biblioteca do Scratch, mantendo a história coerente.

Você deve remixar o seguinte projeto do Scratch no qual o Vibot foi criado por você:

FR:<https://scratch.mit.edu/projects/130549192/>

PT:<https://scratch.mit.edu/projects/306821430/>



É importante compartilhar seu projeto pressionando "Compartilhar" para torná-lo acessível.

Há um formulário un
Google Form onde vc colocará a l'URL
de seu projeto Scratch :

FR: <https://goo.gl/SZ3slV>

PT: <https://forms.gle/SdG2Lqsy2qCVHSv19>



Desafio criativo #3a. O gato Scratch e o Vibot perderam suas cores




Ajude o gato Scratch e o Vibot a encontrarem suas cores usando o comando "cancele os efeitos gráficos" nos códigos dos dois personagens. Em seguida, adicione comandos para fazer os personagens falarem.

Mostra os links para eles Bill?

FR: <https://scratch.mit.edu/projects/122835356>
EN: <https://scratch.mit.edu/projects/144236197>
PT: <https://scratch.mit.edu/projects/306825528/>



(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

- Vibot: "Olá!"
- Gato Scratch: "Oi!"

As cores do Scratch e do Vibot continuam mudando.

- Vibot: "Alguém mudou nossas cores!"
- Gato Scratch: "Você está certo, Vibot!"
- Gato Scratch: "Você pode nos ajudar a encontrar nossas cores originais?"
- Vibot: "Você tem que cancelar os efeitos gráficos."

Agora é sua vez!

As cores voltam ao normal.

- Gato Scratch: "Boa jogada!"
- Vibot: "Obrigado pela sua ajuda."

Aqui
embaixo estão
algumas etapas do
desafio.



Etapas 1. Análise da situação (CTc1sc3).

Você consegue identificar os blocos de código que mudam as cores do gato Scratch e do Vibot?

Etapas 2. Modelagem da situação (CTc4sc2).

Na Biblioteca de código do Scratch, identifique o bloco de código que permitirá cancelar as alterações de cor.

Etapas 3. Continuar o diálogo (CTc4sc2)


Na biblioteca de código Scratch, selecione o bloco de código que permitirá que você programe o seguinte diálogo:

- Gato Scratch: "Boa jogada!"
- Vibot: "Obrigado pela sua ajuda."

Desafio criativo #4a. O Vibot e o gato Scratch mergulham



FR: <https://scratch.mit.edu/projects/123617273/>
EN: <https://scratch.mit.edu/projects/142553826/>
PT: <https://scratch.mit.edu/projects/306825577/>

(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

Reproduza os diálogos e movimentos como eles são indicados abaixo:



O Vibot e gato Scratch fazem mergulho submarino.
O gato Scratch nada na direção do Vibot.

- Gato Scratch: "uhuu, eu amo nadar!"
- Vibot: "Eu também."
- Gato Scratch: "Mas onde estão os peixes?"
- Gato Scratch: "Boa jogada!"
- Vibot: "Obrigado pela sua ajuda."

Agora é só seguir as etapas abaixo:



Etapa 1. Análise da situação.

- Em que lugar devo clicar para adicionar um personagem à conta do Scratch?
- Qual bloco nos permite mover o personagem de um lugar para outro?
- Qual bloco devemos usar para repetir um movimento várias vezes?

Etapa 2. Modelagem da situação.

Na biblioteca de personagens Scratch, escolher ao menos 2 personagens que sabem nadar e colocá-los no fundo.

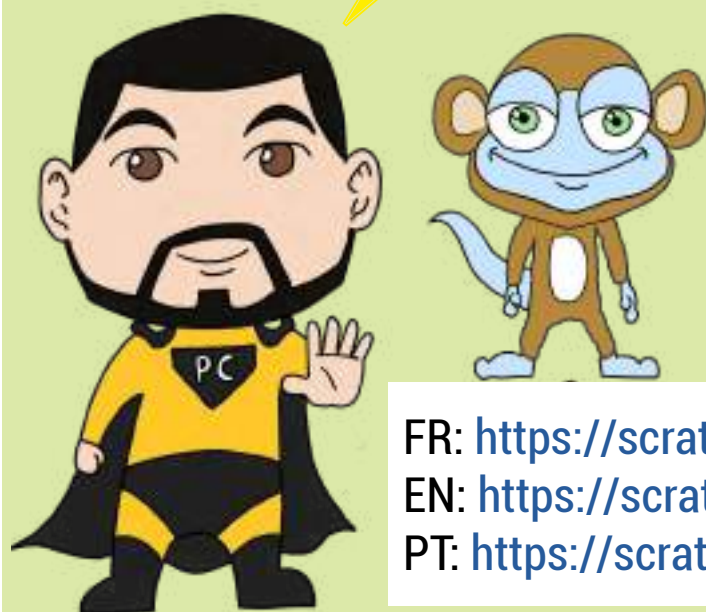
Etapa 3. Adicionar movimentos

À partir da biblioteca do Scratch, escolher os blocos de movimento para mover os novos personagens da direita para a esquerda e da esquerda para a direita.


Etapa 4. Repetir os movimentos

Selecione o tipo de bloco que permite repetir os movimentos quantas vezes desejar. Então, faça os novos personagens se mover 40 vezes.

Desafio criativo #5a. O Vibot e o gato Scratch ajudam o macaquinho



FR: <https://scratch.mit.edu/projects/123266307>
EN: <https://scratch.mit.edu/projects/142608034/>
PT: <https://scratch.mit.edu/projects/306825616/>

(*)  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

- Vibot: "Olá."
- Gato Scratch: "Olá macaquinho! Como você está?"
- Macaquinho: "Eu não estou indo muito bem. Eu queria uma banana, mas não consigo alcançá-la."
- Macaquinho: "Elas estão muito altas!"
- Gato Scratch: "Nós vamos ajudá-lo. Não é Vibot?"
- Vibot: "Certamente."
- Vibot: "Vou ver se sou alto o suficiente para pegá-la."

Vibot se move em direção à árvore.

- Vibot: "Infelizmente, eu não consigo alcançá-la."

O Vibot retorna à sua posição original.

- Gato Scratch: "Eu sei!"
- Gato Scratch: "Eu irei perto da árvore e vou sacudí-la para fazer as bananas caírem!"

Scratch se aproxima da árvore e a sacode. Então, ele retorna à sua posição.

- Gato Scratch: "Isso funciona! Elas estão caindo."

Agora é sua vez!

As bananas caem no chão e o macaquinho se move na direção delas.

- Macaquinho: "Nham, nham! Que prazer! Obrigado amigos."

Etapa 1: Responda as seguintes perguntas nos comentários do programa

- Descreva o algoritmo que permite que a árvore se mova.
- Quais são os blocos de código que permitem que um personagem ou objeto se mova de um lugar para outro?

Etapa 2: Tarefas a fazer no programa

- Corrija o seguinte erro: O gato Scratch não enfrenta seus amigos.
- O gato Scratch deve se mover em direção à árvore (movimento fluido) em vez de aparecer instantaneamente.
- Escolha um ou mais blocos que derrubarão as bananas no chão. Se esforce para tornar o movimento o mais realista
- O conto deve poder ser reiniciado sem erro (ex: objetos permanecem aparecendo ou não são substituídos).
- Na biblioteca de códigos do Scratch, adicione um bloco que permitirá que o macaquinho agradeça seus amigos e insira o texto seguinte:



Macaquinho: "Nham, nham! Que prazer! Obrigado amigos."



Desafio criativo #6a. O Vibot, o gato Scratch e o Gobo jogam futebol



FR: <https://scratch.mit.edu/projects/123266307>
EN: <https://scratch.mit.edu/projects/142608034/>
PT: <https://scratch.mit.edu/projects/306825616/>

(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em !)

O Vibot e o gato Scratch estão com um amigo do gato Scratch, chamado Gobo.

- Gato Scratch: "Olá, Gobo!"
- Gato Scratch: "Vibot, eu apresento a você meu amigo Gobo."
- Vibot: "Olá, Gobo. Meu nome é Vibot. Eu sou um robô".
- Gobo: "Olá amigos!"
- Gobo: "Você quer jogar futebol?"
- Gato Scratch: "Sim!"
- Vibot: "O que é futebol?"

Agora é sua vez!

Gobo diz ao Vibot como jogar futebol.

- Gobo: "É um esporte onde você passa a bola com os pés para fazer gols."
- Vibot: "Ok. Vamos jogar! "

Então, os três amigos passam a bola. A bola vai do gato Scratch para o Gobo para o Vibot e a troca é repetida 5 vezes.

- Vibot: "Eu amo jogar futebol!"

Etapa 1. Análise da situação.

- Quais são os blocos de código que permitem que um personagem se mova de um lugar para outro?
- Qual bloco de código devemos usar para fazer um personagem falar?
- Qual bloco nos permite repetir uma ação quantas vezes desejar?

Etapa 2. Continue o diálogo.

Convidamos você a escolher, na biblioteca do Scratch, o bloco que faz os personagens falar. Em seguida, insira o seguinte diálogo:
Gobo diz ao Vibot como jogar futebol.

- Gobo: "É um esporte onde você passa a bola com os pés para fazer gols."
- Vibot: "Ok. Vamos jogar! "

Etapa 3. Contribua para a situação.

Escolha o bloco apropriado que permitirá que a bola se mova de um jogador para outro (gato Scratch para Gobo, Gobo para Vibot, Vibot para gato Scratch), de acordo com as seguintes coordenadas:

Gato Scratch. x: -125 y: -95

Gobo. x: 1 y: -49

Vibot. x: 62 y: -116

- ▮ Que forma geométrica em que o movimento da bola se realiza?
- ▮ Quais coordenadas devem ser atribuídas a cada personagem para que o triângulo seja isósceles? Equilátero? Escaleno?

Etapa 4. Repita os movimentos (CTc1sc9).

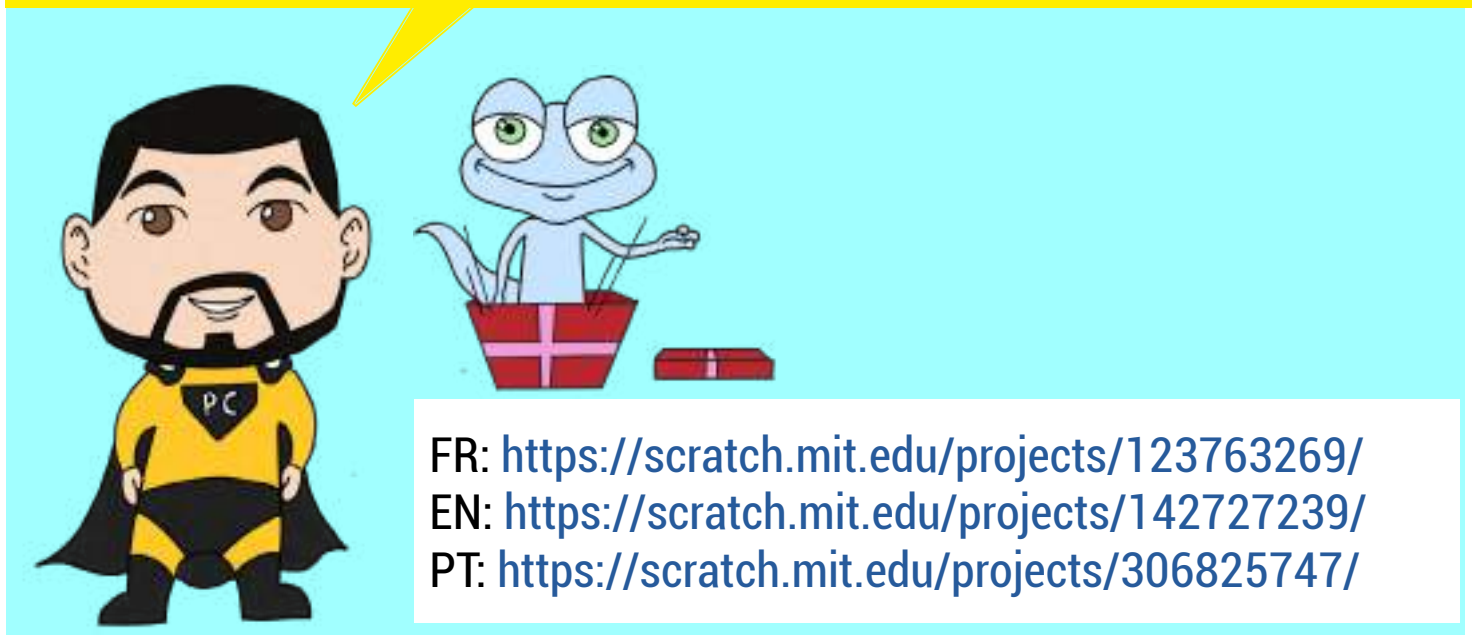
Selecione o tipo de bloco que permite repetir os movimentos quantas vezes desejar. Em seguida, programe a bola para completar o "giro" 5 vezes.


Etapa 5. Encerre o diálogo.

Escolha o bloco que permite falar de personagens. Em seguida, insira as seguintes palavras:

- Vibot: "Eu amo jogar futebol!"

Desafio criativo #7a. O gato Scratch prepara uma surpresa para o Vibot



(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

O gato
Scratch está preparando uma festa
surpresa para o Vibot. Na sala decorada estão
todos os amigos do Vibot.



- Gato Scratch: "Olá!"
- Todos os amigos: "Olá! Tudo bem? Oi! Olá! "
- Gato Scratch: "Estamos organizando uma festa surpresa para o aniversário do Vibot."
- Gato Scratch: "Em alguns segundos, todos os amigos irão desaparecer."
- Gato Scratch: "Então, quando o Vibot chegar, os amigos

sairão de seus esconderijos".

- Gobo: "Eu ouço Vibot. Depressa, vamos nos esconder! "

Os amigos de Vibot desaparecem, mas o gato Scratch permanece visível. O Vibot aparece e se move até o gato Scratch.

- Gato Scratch: "Vibot!"

O gato Scratch envia uma mensagem ("surpresa") para todos os amigos para dizer que é hora de aparecer.

Agora é sua vez!

Quando os amigos recebem a mensagem ("surpresa") do gato Scratch, aparece a ponta do nariz deles.

- Todos: "Surpresa !!"
- Gato Scratch: "Feliz aniversário, meu amigo!"
- Vibot: "Uau! Obrigado amigos. Vocês são tão legais. Que surpresa legal! Obrigado!"

Etapa 1. Análise da situação.

- Que bloco de aparência permite mostrar um personagem que estava escondido?
- Qual bloco de código controla a espera de um personagem por mais de 1 segundo antes de dizer ou fazer alguma coisa?
- Como você faz um personagem dizer alguma coisa?

Etapa 2. Modelagem da situação.

Escolha o bloco de aparência que fará aparecer, quando receber a mensagem "surpresa", os seguintes personagens: Giga, Pico, Tera, Gobo, Monkey2, Crab e Nano. Em seguida, adicione um comando de controle para que os personagens esperem 1 segundo.

Etapa 3. Continue o diálogo.


Na biblioteca de códigos do Scratch, adicione um bloco com o qual o Giga, o Pico, o Tera, o Gobo, o Monkey2, o Crab, o Nano e o gato Scratch poderão surpreender o amigo com o seguinte texto:

- Giga, Pico, Tera, Gobo, Monkey2, Crab, Nano e gato Scratch: "Surpresa!!"
- Gato Scratch: "Feliz aniversário, meu amigo!"
- Vibot: "Uau! Obrigado amigos. Vocês são tão legais. Que surpresa legal!"

Desafio criativo #8a. O gato Scratch e o Vibot perdem seu "fundo"



FR: <https://scratch.mit.edu/projects/126047102>
EN: <https://scratch.mit.edu/projects/142737514>
PT: <https://scratch.mit.edu/projects/306825816/>

(*)  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

O gato Scratch e o Vibot estão na praia e aproveitam o clima agradável.

- Gato Scratch: "Ahh!"
- Gato Scratch: "Que dia perfeito para tomar sol na praia".
- Vibot: "Sim, mesmo que eu não possa me bronzear."

De repente, os amigos se encontram nas pistas de esqui.

- Gato Scratch: "Mas o que está acontecendo?"
- Gato Scratch: "Onde estamos?"
- Vibot: "Brrr. Estou com frio."
- Vibot: "Os robôs não são feitos para viver frio!"

Os amigos são transportados longe das pistas de esqui e ... no espaço!

- Vibot: "Eu não entendo o que está acontecendo conosco!"
- Gato Scratch: "Acho que alguém está tentando nos enganar."
- Vibot: "Enviando-nos para o espaço?"

O gato Scratch e o Vibot se encontram em frente a um castelo na Idade Média.

- Vibot: "E na Idade Média?"
- Gato Scratch: "Sim!"
- Gato Scratch: "Alguém mudou nossos planos de fundo."

Um dragão cuspidor de fogo está perigosamente perto do gato Scratch.

- Vibot: "Gato Scratch, cuidado!"

Os amigos desaparecem novamente para reaparecer em um palco na frente de uma multidão.

- Gato Scratch: "Oh, ai!"
- Vibot: "Hum, Scratch. O que estamos fazendo?"
- Gato Scratch: "Alguém tem que colocar de volta o fundo da praia !!"

Agora é sua vez de jogar!

O microfone do Vibot e a guitarra do gato Scratch desaparecem. Os dois amigos se encontram na praia novamente, usando seus óculos escuros.

- Gato Scratch: "Ufaa. Que aventura!"
- Vibot: "Sim, mas estou feliz por estar de volta."
- Gato Scratch: "Eu também, obrigado por nos tirar dessa bagunça."

Etapa 1. Análise da situação.

- Onde devo clicar para ir para gerenciar os diferentes fundos de tela?
- Como podemos mudar um fundo?
- Qual bloco de código "de evento" deve ser usado para alternar de um fundo para outro ao receber uma mensagem?
- Que blocos de código "aparência" você precisa usar para fazer com que personagens diferentes desapareçam e apareçam novamente quando trocamos de um fundo para outro?

Etapa 2. Modelagem da situação.

No código do fundo de tela, adicione os blocos que permitirão que o plano de fundo mude quando a mensagem "end-spotlight-stage" for enviada.

Etapa 3. Modelagem da situação.

No código dos personagens para o microfone e o violão, adicione o bloco apropriado que os fará desaparecer quando o fundo de tela mudar para a cena da praia. Além disso, será necessário escolher o bloco que fará aparecer os óculos e o guarda-sol quando o fundo de tela mudar para a cena da praia.

Etapa 4. Continue o diálogo.

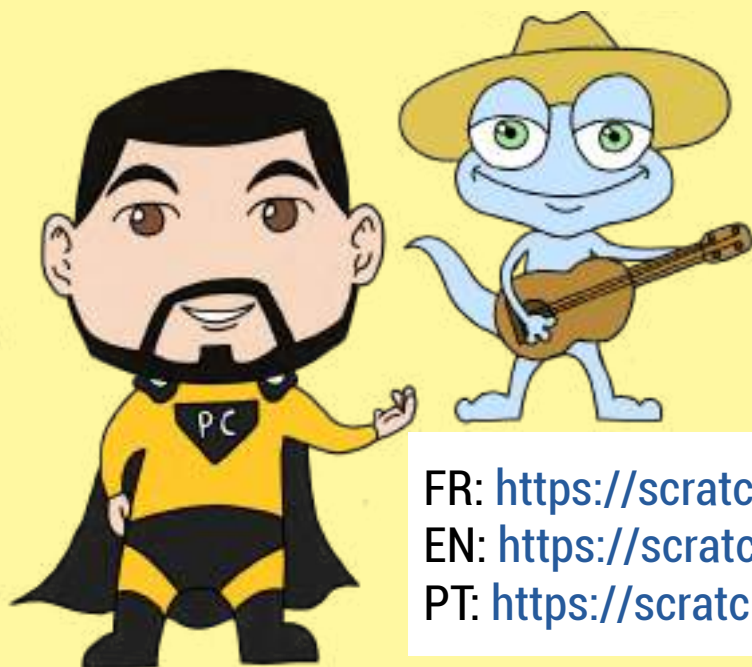
Na biblioteca de códigos do Scratch, adicione um bloco com o qual o gato Scratch e o Vibot podem continuar diálogo com o seguinte texto:

- Gato Scratch: "Ufaa. Que aventura!"
- Vibot: "Sim, mas estou feliz por estar de volta."
- Gato Scratch: "Eu também, obrigado por nos tirar dessa bagunça."



Seguindo esses passos fica fácil não é ?



Desafio criativos #9a. O Vibot e o gato Scratch formam um grupo de música



FR: <https://scratch.mit.edu/projects/126847033/>
EN: <https://scratch.mit.edu/projects/142792129>
PT: <https://scratch.mit.edu/projects/306825942/>

(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em !)

Vou te ajudar,
aqui estão algumas
etapas para resolver este
desafio!



Etapas 1. Análise da situação (CTc3sc2).

- Qual algoritmo nos permite tocar um instrumento quando um personagem (gato Scratch, por exemplo) encosta no instrumento em questão?
 - ▮ Qual bloco de código pode adicionar um som a um personagem? Como podemos criar uma melodia?

Etapa 2. Modelagem da situação(CTc3sc4).

- Escolha o algoritmo apropriado para que a guitarra toque quando estiver em contato com o gato Scratch ou o Vibot.
- Então, escolha os sons para criar uma melodia.

Etapa 3. Modelagem da situação (CTc5sc6).

- Adicione um 5º instrumento e faça-o tocar a melodia de sua escolha pelo personagem de sua escolha.

Etapa 4. Mudando a situação(CTc3sc6).

- O Vibot também gostaria de poder tocar violão. Modifique o programa para fazer com que o Vibot possa tocar a guitarra. Como podemos fazer para que o gato Scratch e o Vibot possam tocar ao mesmo tempo?

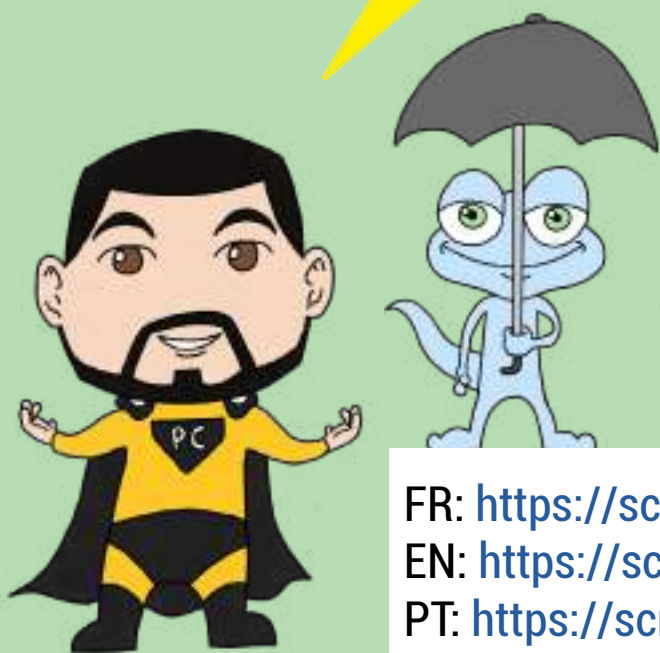
Etapa 5. Altere a situação.

- Crie músicas com o gato Scratch e o Vibot.
- Se você quiser personalizar seu grupo, você pode gravar sua voz e adicioná-la ao seu projeto Scratch. Você pode então usar os algoritmos apropriados para fazer seu som de voz quando um dos dois caracteres (ou um terceiro?) colide com o microfone no centro da sala.


E aí, conseguiu?
Então vamos para o
próximo desafio!



Desafio criativo #10a. O gato Scratch, o Vibot e as estações



FR: <https://scratch.mit.edu/projects/132948199>
EN: <https://scratch.mit.edu/projects/142785109/>
PT: <https://scratch.mit.edu/projects/306825968/>

(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

Este desafio tem 4 fundos de tela, um para cada estação do ano. Cabe a você determinar o que acontece durante cada estação. Cada estação deve conter uma história curta relacionada à estação. No entanto, você deve atender a certas restrições.



Etapa 1. Modelagem

Antes de começar a programar, organize suas ideias de maneira lógica, escrevendo-as em um pedaço de papel ou em um documento do GoogleDocs. Neste documento, escreva os elementos (personagens, blocos de código, algoritmos, diálogos, etc.) Planeje o que você precisará para cada estação, levando em conta as restrições.

Etapa 2. Restrições

- Primavera: Na primavera, as flores crescem. Adicione uma flor que cresça gradualmente (isto é, a flor não deve aparecer de um momento para outro, ela deve crescer diante de nossos olhos).
- Verão: Adicione pelo menos 2 outros personagens para fazer companhia para o Scratch e o Vibot.
- Outono: insira folhas de árvores que caem continuamente.
- Inverno: Como é inverno, você pode adicionar flocos de neve que caiam continuamente. Além disso, o gato Scratch e o Vibot constroem um boneco de neve. Certifique-se de que haja uma progressão na construção do boneco de neve (em outras palavras, o boneco de neve não deve aparecer de uma hora para a outra já construído).
- Para adicionar realismo, certifique-se de que os personagens estejam vestidos apropriadamente para a estação em que estão.

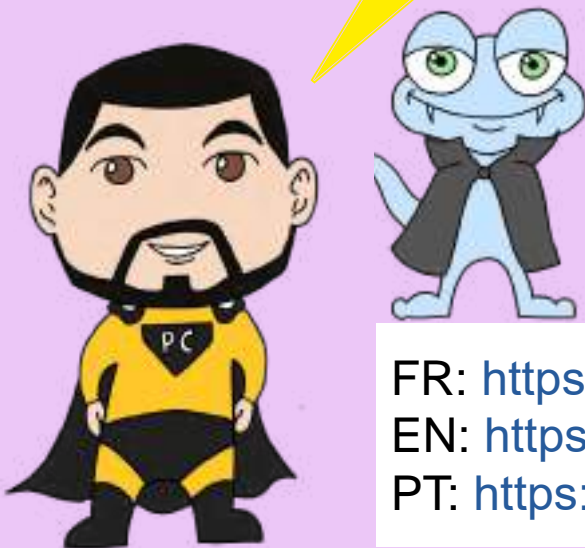
Etapa 3. Incluir Transições

Para cada estação, há um personagem com o nome da estação formado por letras coloridas. Use os blocos de código que permitirão que você exiba os nomes das estações assim que mostrar o fundo de tela apropriado. Embora os efeitos visuais já tenham sido aplicados aos nomes das estações, você pode alterá-los para adicionar os efeitos de sua escolha.


Etapa 4. Seja criativo!

Dê asas a sua criatividade você pode ir além das restrições, você pode fazer o que quiser com cada estação. Por exemplo, uma estação pode ser um jogo enquanto outra estação poderia ser uma história. Divirta-se!

Desafio criativo #11a. O gato Scratch vai ao Halloween



FR: <https://scratch.mit.edu/projects/128340027>
EN: <https://scratch.mit.edu/projects/144254480>
PT: <https://scratch.mit.edu/projects/306826007/>

(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

Etapa 1. Análise da situação(CTc3sc2)

Em uma folha de papel, responda às seguintes perguntas e explique, em suas palavras, o que acontece em cada situação:

- Qual algoritmo pode mover aleatoriamente um personagem? Como podemos fazê-lo realizar um movimento preciso em um loop (por exemplo, fazê-lo desenhar um quadrado)?
- Qual algoritmo devemos usar para criar um efeito quando há uma colisão entre dois personagens (por exemplo, dois personagens se batendo?).

Etapa 2. Complete o programa (CTc3sc4)

- Escolha o algoritmo apropriado para mover o morcego por um caminho de sua escolha (forma geométrica).
- Certifique-se de fazer o Scratch perder um ponto para quando houver uma colisão entre ele e o morcego.

Etapa 3. Corrigir o programa (CTc6sc1 et CTc6sc2)

- Há um erro no programa (no nível 2). Identifique e corrija este erro para que possamos jogar sem problemas.

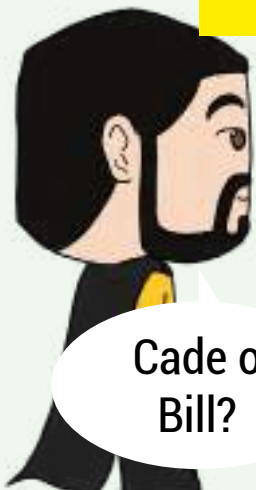
Etapa 4. Modificar o Programa(CTc5sc6)

- Encontre uma maneira de tornar o jogo (níveis 1 e 2) mais difícil.

Etapa 5. Expanda o programa (CTc5)

- Adicione o nível 3 ao programa. Encontre um fundo de tela relacionado ao tema e dê a ele um nome adequado.
- Certifique-se de que a transição do Nível 2 para o Nível 3 seja suave e eficiente.
- Adicione um terceiro obstáculo, mantendo o fantasma e o morcego.


Desafio criativo #12a. O gato Scratch, o Vibot, e jogo de fuga



Cade o
Bill?

FR: <https://scratch.mit.edu/projects/128340027>
EN: <https://scratch.mit.edu/projects/144254480>
PT: <https://scratch.mit.edu/projects/306826007/>



(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

Etapa 1. Análise da situação

- Qual é o objetivo do jogo e como nós poderemos ganhar?
- É possível encontrar pistas sem clicar em todos os lugares?

Etapa 2. Entenda a situação

- Como podemos definir o que é um código?
- Um código pode ser formulado de diferentes maneiras (programação, letras, números, cores, símbolos, etc.)?
- Como um determinado código pode levar à realização de uma determinada ação?
- Neste jogo, há uma hierarquia importante nos diferentes eventos que podem ocorrer (por exemplo, se clicarmos em A, B aparece). Como explicar essa hierarquia na linguagem normal e na linguagem Scratch?
- Há uma ou mais colisões entre dois personagens? Se sim, quais são suas funções?

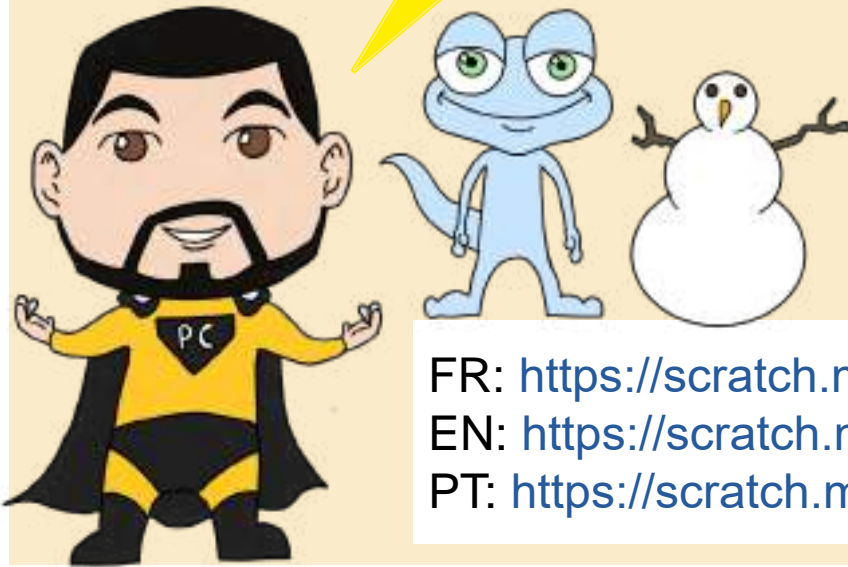
Etapa 3. Corrigir o código

- Há uma inconsistência na sala 2, você pode encontrá-la? Você pode corrigi-la para que o movimento esperado seja o mais natural possível?


Etapa 4. Complete o programa

- Complete o programa para criar outra forma de codificar para que o gato Scratch e o Vibot possam sair da segunda sala.
- Adicione personagens e esconda suas pistas na sala.
- Crie o seu próprio jogo de fuga.

Desafio criativo #13a. O gato Scratch e o Vibot fogem das bolas de neve



FR: <https://scratch.mit.edu/projects/128340027>
EN: <https://scratch.mit.edu/projects/144254480>
PT: <https://scratch.mit.edu/projects/306826007/>

(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

Etapa 1. Análise da situação (CTc3sc2)

- Qual algoritmo pode mover aleatoriamente um personagem? Como podemos fazê-lo fazer um movimento repetitivo (por exemplo, fazê-lo fazer um quadrado)?
- Qual algoritmo devemos usar para criar um efeito quando há uma colisão entre dois personagens (dois personagens se batendo)?
- Como instalar um sistema de pontos que determinará se o Scratch perde ou ganha o jogo?

Etapa 2. Modelar a situação

- Em um pedaço de papel ou em um documento do Google Docs, registre as diferentes formas para ter sucesso no jogo (algoritmos de movimento, função de transmissão para um personagem envie uma mensagem para outro personagem, sistema de pontos, etc. .)
- Depois que seus códigos foram claramente analisados e modelados, você pode passar para o estágio de programação.

Etapa 3. Complete o programa (CTc3sc4)

- Encontre os algoritmos que permitirão que os três personagens se movam de um lugar para outro (o gato Scratch se moverá em resposta a um comando do jogador e o Pinguim e o Vibot se moverão aleatoriamente ou num caminho que você tenha predefinido).
- Encontre um jeito do Pinguim jogar bolas de neve para seus amigos. Quando há uma colisão entre uma bola de neve e o gato Scratch, o gato Scratch perderá um ponto.
- Estabelecer um sistema de pontos. Determine o número de pontos necessários para vencer e, o inverso, a quantidade de bolas de neve que se baterem no gato Scratch o farão perder o jogo.
- Certifique-se de que os movimentos dos personagens são fluidos e naturais. Lembre-se que o programa deve ser reiniciado sem problemas.

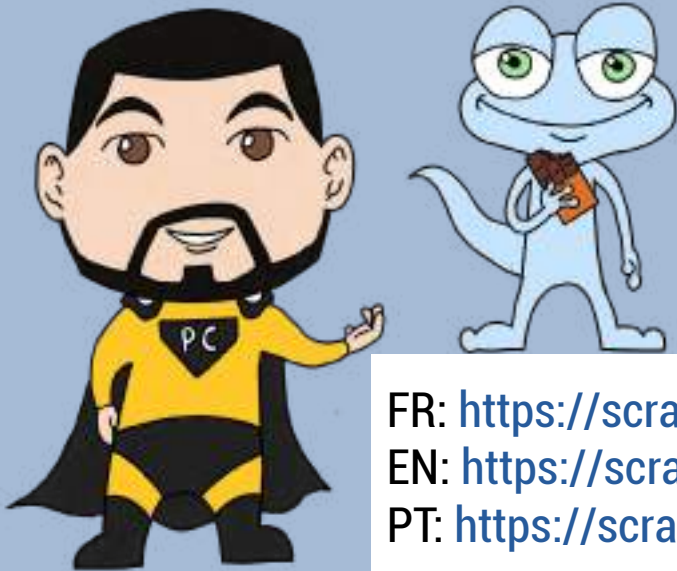
Sugestões:

- Você pode adicionar dificuldades diferentes se considerar seu jogo muito fácil.
- Você será capaz de criar vários níveis, incorporando novos obstáculos em cada nível.
- Você pode adicionar um abrigo (por exemplo, uma árvore ou uma montanha de neve) atrás do qual o gato Scratch estará seguro por um certo período de tempo.
- Por que não adicionar uma melodia/música para melhorar o humor do seu jogo?

Esse aí foi fácil! Vamos para o próximo?



Desafio criativo #14a. O Vibot e os chocolates



FR: <https://scratch.mit.edu/projects/133059181/>
EN: <https://scratch.mit.edu/projects/146235612/>
PT: <https://scratch.mit.edu/projects/306826070/>

Neste desafio, o Vibot faz uma caça aos ovos de Páscoa! Vários ovos foram escondidos em um local de sua escolha (o pátio de sua escola, sua casa, sua sala de aula, etc). Cabe a você determinar como a cena se desenrola e ajudar o Vibot a encontrar os chocolates que ele tanto ama.



Etapas 1. Análise da situação

- Como podemos adicionar fundo de tela de nossa escolha no Scratch?
- Como podemos ter objetos ocultos e quais blocos de código nos permitem fazê-los aparecer?
- Como podemos criar nossos próprios personagens?
- Quais blocos de código nos permitem criar uma sucessão de eventos? (Por exemplo, se o Vibot encosta em A, A envia uma mensagem para B e C, que por sua vez fazem algum movimento).

Etapas 2. Modelagem

- Em um documento do GoogleDocs, identifique os diferentes elementos (tela de fundo, personagens, blocos de códigos, diálogos, etc) necessários para concluir seu desafio.
- Pode ser útil desenhar um esboço, em uma folha de papel, da cena que você gostaria de criar. Desta forma, você pode determinar com antecedência onde você pode esconder seus ovos.
- Quão importante é essa preparação antes de começar a programar?

Etapas 3. Escolha um plano/tela de fundo

- Escolha um lugar na sua vida diária que lhe interessa. Este lugar pode ser sua sala de aula, sua casa, a escola, a sua rua, etc. Certifique-se que a decoração do cenário é grande o suficiente para o Vibot se movimentar para encontrar os ovos.
- Uma vez que sua localização ideal tenha sido escolhida, tire uma foto dela e insira a foto no Scratch.

Etapas 4. Criando e inserindo personagens

- Crie o seus próprios personagens/ovos e pode colorir como quiser.
- Então encontre uma maneira de esconder os ovos na decoração do cenário. Isso pode ser feito de várias maneiras diferentes. Quanto mais ovos você esconder de maneiras diferentes, melhor.
- Você pode incluir outros personagens em seu desafio, como itens que ajudarão a decorar, ou personagens que ajudarão na caça ao Vibot.



Seja criativo!

Etapa 5. Criando um Cenário Completo

- Determine como a caça aos ovos de Páscoa do Vibot se desenvolverá: será na forma de um jogo, no qual o jogador terá que manipular o Vibot para ajudá-lo a encontrar seus ovos, ou estará sob forma de história/conto, onde cada elemento do desafio terá sido programado por você antecipadamente? Sinta-se à vontade para se inspirar em outros desafios deste guia.
- Pense em integrar diálogos em seu desafio.

Variações

- Se você está pensando em criar um jogo, seria interessante usar um sistema de pontos (blocos de código, "dados" no Scratch).
- Se você se sentir confortável o suficiente, será possível adicionar várias cenas ao desafio. Assim, você pode tirar algumas fotos que representam sua vida diária e ocultar os ovos de chocolate nas diferentes cenas.
- O Scratch permite incluir sons e músicas em seus projetos. Seria possível adicionar um som de vitória cada vez que o Vibot encontrasse um ovo. Você também pode colocar uma música de sua escolha, para dar uma impressão ainda melhor sobre sua vida diária.

Desafio criativo #15a. O Vibot e as peças de seu carro



Esse desafio é para
você!
Tente você construir o projeto e
compartilhar o link!

Desafio criativo #16a. O gato Scratch fala sobre o Papai Noel ao Vibot



Neste desafio, o Vibot e o gato Scratch estavam brincando na neve. De repente, um objeto cai do céu. O Vibot acha que é um objeto, mas é o Papai Noel. O gato Scratch tenta descrever o que é o Natal para o Vibot. Você continuará a história e codificará no Scratch.

Tente você construir o projeto e
compartilhar o link!

Gato Scratch e o Vibot brincam na neve.

- Gato Scratch: "Huhuu."
- Gato Scratch: "Olhe Vibot, eu estou criando um telefone com a neve."
- Vibot: "Olha, isso é legal! Eu criei um carro!"

De repente, algo cai do céu na frente do Vibot.

- Vibot: "Venha ver gato Scratch, um objeto caiu do céu?"

O gato Scratch se move para o Vibot

- Gato Scratch: "Hoo, é o Papai Noel!"
- Vibot: "O que é o Papai Noel e o que é o Natal?"

Continue o diálogo contando ao Vibot o que é o Natal e o Papai Noel usando o personagem do gato Scratch.

Aqui embaixo estarão as etapas a serem seguidas.



Etapas 1. Análise da situação (CTc3sc2)

- Como podemos adicionar uma tela de fundo com neve no Scratch?
- Como os objetos podem cair do céu?
- Quais são os personagens que são necessários para programar este desafio?
- Quais blocos de código nos permitem criar uma sucessão de eventos?

Etapa 2. Modelagem

- Em um documento do Google Docs, identifique os diferentes elementos (tela de fundo, personagens, blocos de códigos, diálogos, etc) necessários para concluir seu desafio.
- Pode ser útil procurar fundos e fotos do Papai Noel na internet.
- Quão importante é essa preparação antes de começar a programar?

Etapa 3. Escolha um plano/tela de fundo

- Escolha um cenário/fotos de natal que te interessa.
- Uma vez que sua localização ideal tenha sido escolhida, insira ela no Scratch.

Etapa 4. Criando e inserindo personagens

- Crie o seus próprios personagens além do Papai Noel e pode colorir como quiser.
- Seja criativo!

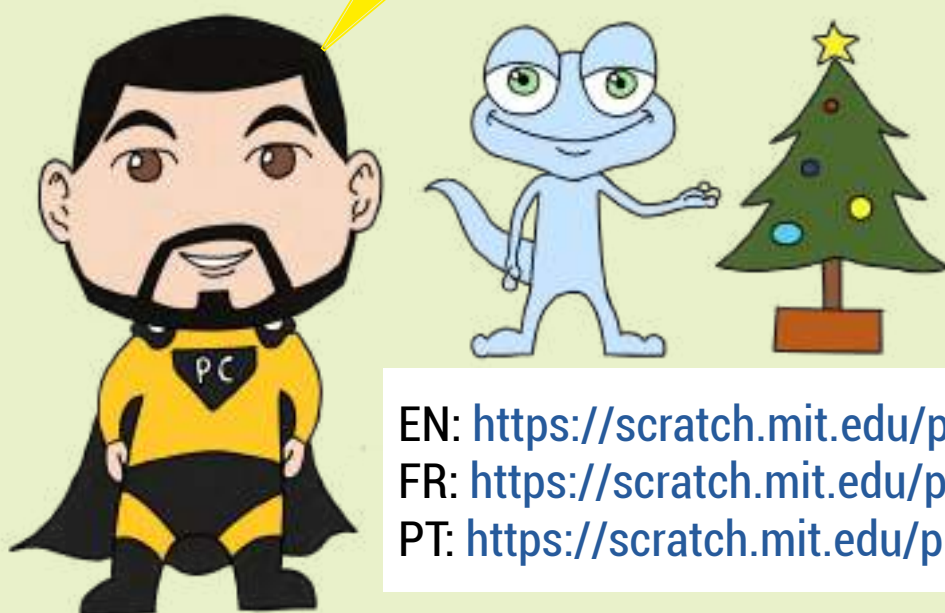
Etapa 5. Criando um Cenário Completo

- Determine como o gato Scratch irá descrever o Natal para o Vibot? Sinta-se à vontade para se inspirar em outros desafios deste guia.
- Pense em integrar diálogos em seu desafio.


Agora vamos para mais um desafio.



Desafio criativo #16b. O Vibot e o gato Scratch festejam o Natal



EN: <https://scratch.mit.edu/projects/146251411/>
FR: <https://scratch.mit.edu/projects/137697840/>
PT: <https://scratch.mit.edu/projects/306826152/>

( Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

Etapa 1. Análise da situação (CTc3sc2)

- Que tipo de sala é essa?
- Qual é o tema da cena em que o gato Scratch e o Vibot estão? Podemos adicionar elementos que se encaixem no tema?
- Existe um objeto que não se encaixa no tema; você consegue identificá-lo? Se sim, você pode fazer o objeto se tornar relacionado ao tema, sem fazê-lo desaparecer? Seja criativo!

Etapa 2. Adicionando elementos festivos

- O gato Scratch e o Vibot comemoram o Natal: Adicione suas roupas adequadas!
- O que poderia ser colocado acima da lareira?
- Adicione decorações à sala onde quiser.

Etapa 3. Animar a fonte de calor

- Há chamas na chaminé, mas estão imóveis. Encontre uma maneira de manter as chamas de uma maneira natural.

Etapa 4. Decore a árvore!

- Usando os diferentes itens nas caixas, ajude o gato Scratch e o Vibot a decorar a árvore de Natal.
- Você pode adicionar itens que não estão nas caixas. Seja criativo; a árvore deve estar ao seu gosto!
- Está faltando um objeto para colocar no topo da árvore ...

Etapa 5. Embale os presentes

- Existem duas caixas à direita da sala; embale-as e decore-as.

Etapa 6. Crie a atmosfera!

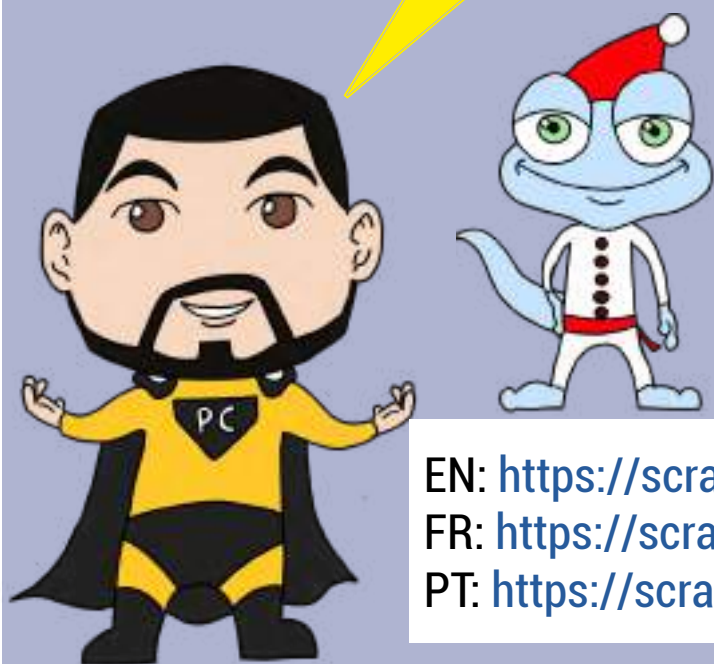
- Adicione algumas músicas especiais para criar uma atmosfera agradável para o gato Scratch e o Vibot.
- Por que não adicionar um som de fogo?

Etapa 7. Crie uma linda história.


Crie um cenário no qual o gato Scratch e o Vibot ofereçam presentes de Natal. Seu cenário deve incluir:

- Diálogos.
- Movimentos realistas (viajar, um amigo que dá e um amigo que recebe um presente, um amigo que desembrulha uma caixa e aparece o presente).

Desafio criativo #17a. O gato Scratch e o Vibot vão ao carnaval do Quebec



EN: <https://scratch.mit.edu/projects/146251411/>
FR: <https://scratch.mit.edu/projects/137697840/>
PT: <https://scratch.mit.edu/projects/306826152/>

(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#)!)

Neste desafio, o gato Scratch e o Vibot vão para o Carnaval de Quebec, onde encontram várias atividades para fazer. O Carnaval oferece a eles a oportunidade de deslizar na neve, patinar, construir um palácio de gelo, fazer doce de açúcar e fazer anjos na neve. Embora cada atividade tenha sido “colocada” no projeto Scratch, cabe a você adicionar os elementos essenciais ao sucesso das atividades. Siga as instruções para dar ao Vibot e ao gato Scratch um Carnaval inesquecível!



Atividade 1. Deslizar na Neve

- O Vibot e o gato Scratch querem deslizar, mas eles não têm uma prancha.. Cabe a você construir uma!
- Em seguida, adicione os blocos de movimento necessários para a descida. Certifique-se de que o movimento de descida é natural.

Atividade 2. Patinação no Gelo e Hóquei no Gelo

- Você é solicitado a reparar alguns erros de programação. O primeiro erro ocorre quando o personagem Pico patina na neve, o que não é realista. A segunda é quando o personagem Pico dá um passo para trás, o que não é um movimento muito natural.
- Para adicionar realismo ao desafio, certifique-se de que o corpo do Pico está na direção certa quando ele está patinando.
- Além disso, você pode adicionar caracteres (2 a 4) e fazer com que eles joguem um jogo de hóquei. Eles precisarão de instrumentos (paus, puck, gols, etc.). Certifique-se de incluir o movimento dos paus e um movimento para o disco.

Atividade 3. Construindo um Palácio de Gelo

- Nesta atividade, você é solicitado a construir um palácio de gelo usando um bloco de gelo disponível (dica: você pode duplicar o bloco para ter mais de um). Sinta-se à vontade para ajustar o tamanho blocos como desejar!).

Atividade 4. Doces de açúcar na neve

- Você tem à sua disposição uma caldeira em que é uma deliciosa calda de açúcar. Cabe a você fazê-lo borbulhar/mexer (mais uma vez, pedimos que o movimento seja natural) e adicionar movimentos ao bastão de madeira para que ele possa obter os puxões (para formar os doces, parecido com maçã do amor, enrola o açúcar no bastão de madeira)>

Atividade 5. Anjos na neve

- Nesta atividade muito simples, você precisa adicionar movimentos ao gato Scratch para que ele possa desenhar um anjo na neve. Obviamente, os movimentos precisam modificar, de certa forma, a neve na qual o gato Scratch está deitado.

Atividade 6. Atividade secreta e oculta

- Cabe a você encontrá-la ...

Desafio criativo #18a. O Vibot, o gato Scratch e o esporte nacional canadense

Neste desafio, o Vibot, o gato Scratch e seus amigos fazem parte de um jogo de hóquei de mesa. Você é o time azul e controla o gato Scratch (WASD), Pico (flechas) e Nano, o goleiro (barra de espaço).



EN: <https://scratch.mit.edu/projects/146655561/>
FR: <https://scratch.mit.edu/projects/136979543/>
PT: <https://scratch.mit.edu/projects/306826337/>

Etapa 1. Jogue!

- Teste o jogo. Para ganhar, você tem que marcar 5 gols.
- Tente identificar os principais problemas ou “bugs” contidos no programa.



Etapa 2. Modelagem

- Em um pedaço de papel, faça uma lista dos problemas a serem resolvidos.
- Para cada problema, explique com suas palavras como você vai resolver o problema.
- Em seguida, explique na linguagem Scratch (ou seja, explicando quais blocos e algoritmos serão usados) como você vai fazer para resolver o problema.
- Tão logo os problemas e soluções sejam identificados, comece a fazer alterações no programa.

Etapa 3. Modificar o Programa


- Quatro personagens não se movem. Inspirado pelos outros personagens, faça movimentos para o Vibot, o Snowman, o Dino e o Mr. Creature. Certifique-se de que os personagens não possam sair de suas “linhas”.
- Adicione uma introdução ao programa (por exemplo, uma introdução que inclua um botão “Iniciar”).
- Precisa-se de cinco pontos para ganhar o jogo. Adicione um algoritmo que faça o jogo parar quando o jogador ou o computador marcar 5 pontos.
- A transição entre um gol e o retorno ao jogo não é natural. Certifique-se de alterar esta transição para que haja uma pequena parada entre o gol e o retorno ao jogo.
- Outros erros aparecerão. Ajuste-os para que o programa se desenrole de acordo.

Desafio criativo #18b. O Vibot, o gato Scratch e o esporte nacional canadense (sua vez de programar!)



Crie o seu próprio jogo de hóquei de mesa.

EN: <https://scratch.mit.edu/projects/146655561/>
FR: <https://scratch.mit.edu/projects/136979543/>
PT: <https://scratch.mit.edu/projects/306826337/>

(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

Crie um jogo como "Air hockey". A tela de fundo já está no programa, mas você pode criar um novo documento do Scratch para começar de novo ou alterar a tela de fundo que já está lá. O jogo terá um disco e duas pás com os quais você pode marcar gols.

Seu jogo deve incluir algumas restrições:

- As pás não podem ir além da linha central.
- O disco deve saltar quando colide com as pás ou as laterais do ringue.
- A velocidade do disco deve reduzir gradualmente se não colidir com as pás ou com os lados do ringue.
- A pá/raquete do jogador deve ser controlada com o mouse. Você também pode criar um jogo que exija duas pessoas. Nesse caso, você pode considerar usar as teclas do teclado (WASD e setas) para mover as pás.
- O computador deve estar constantemente se movendo em direção ao disco na tentativa de acertar o disco.


- Para os algoritmos que permitem que o disco e as pás se movam, não hesite em consultar o Desafio# 18a.
- Você pode adicionar diferentes níveis de dificuldade. Existem várias maneiras de fazê-lo (dica: aumentar a velocidade do disco).

Desafio criativo #19a. O Vibot em Londres



Neste curto desafio, o Vibot quer visitar Londres. Para fazer isso, você deve ajudá-lo a entrar no ônibus e levá-lo para a estrada.

FR / EN: <https://scratch.mit.edu/projects/122831125/>
PT: <https://scratch.mit.edu/projects/306826368/>

(*  Antes de iniciar a programação do desafio, não se esqueça de remixar o projeto clicando em [Remix](#) !)

Etapa 1. Ajude o Vibot a embarcar no ônibus

- Quais blocos de código permitem que um personagem se mova naturalmente de um ponto para outro?
- Use os códigos que permitirão que o Vibot se mova de sua posição inicial para o ônibus e entre nele.
- Certifique-se de que o movimento do Vibot seja o mais natural possível (por exemplo, o que acontece quando alguém se afasta de nós? Essa pessoa mantém o mesmo tamanho?)
- Como podemos tornar possível ver o Vibot apenas pelas janelas do ônibus?

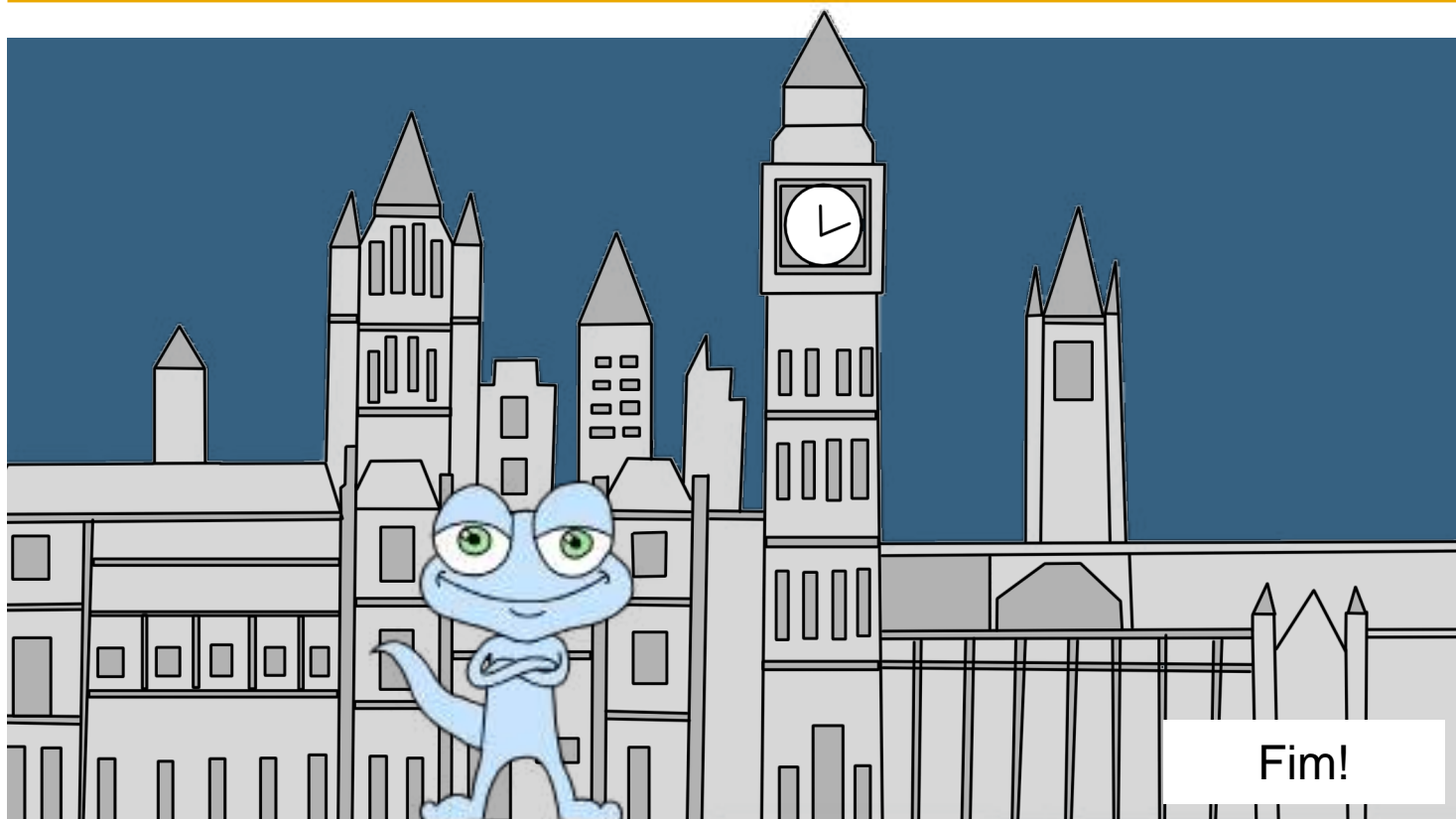
- Adicione outros personagens/passageiros no ônibus. Ou melhor ainda tire um “selfie” e vá passear com o Vibot adicionando sua foto no Scratch!

Etapa 2. Faça o ônibus avançar

- Quais códigos devem ser usados para garantir que o ônibus comece a se mover assim que o Vibot entrar no ônibus? Qual algoritmo devemos usar para criar um efeito quando há uma colisão entre dois personagens (dois personagens se encostando)? Existe uma maneira de adicionar um atraso entre a colisão e a reação?
- Mova o ônibus de sua localização original para a outra extremidade da imagem.
- Certifique-se de que o ônibus não se mova muito rápido e seja natural. Por exemplo, você pode girar as rodas do ônibus durante a viagem?

Etapa 3. Seja criativo!

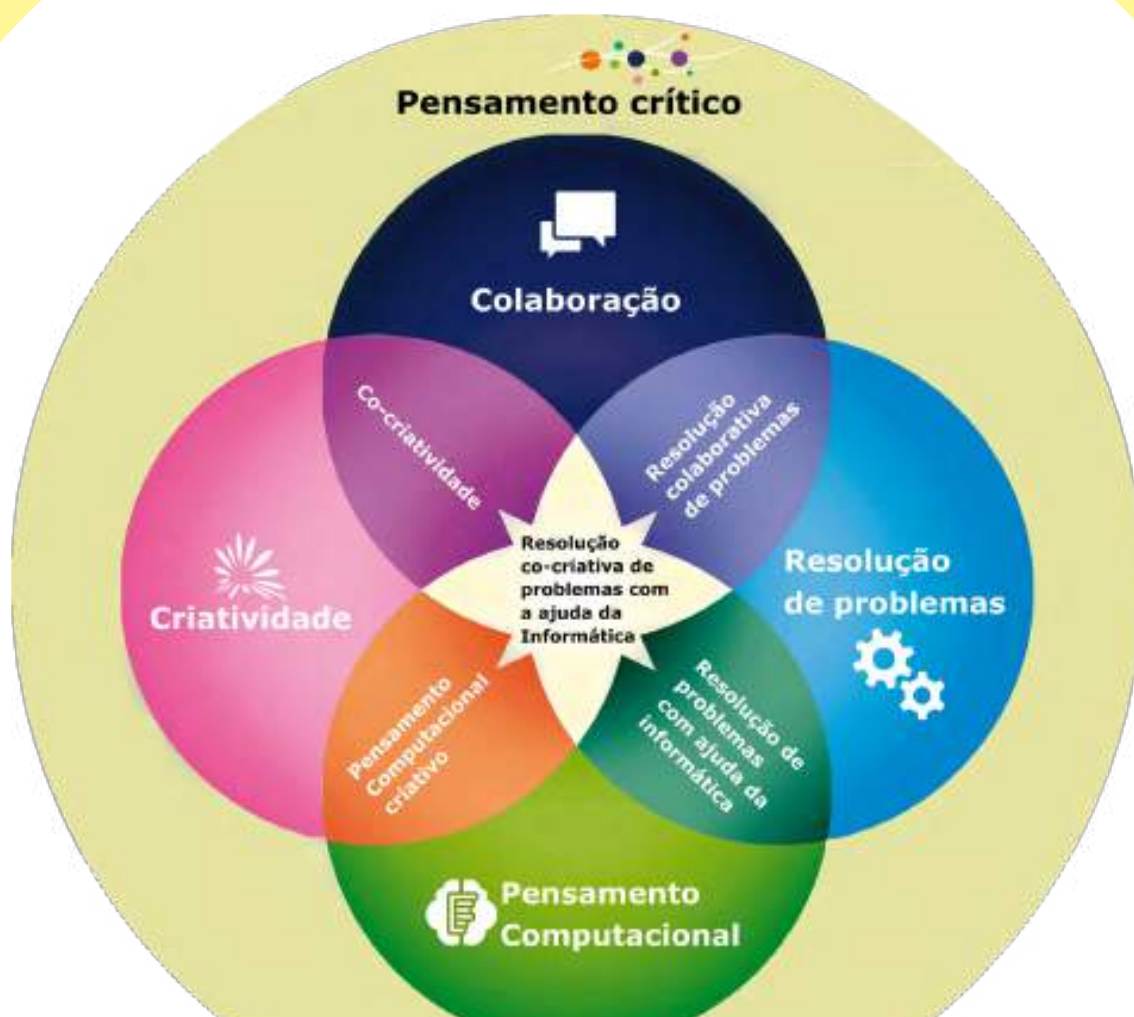
- Divirta-se! Você pode adicionar outros personagens, objetos ou paisagens para ampliar o percurso do Vibot.





Pistas para avaliação dos desafios segundo as Competências para o século 21

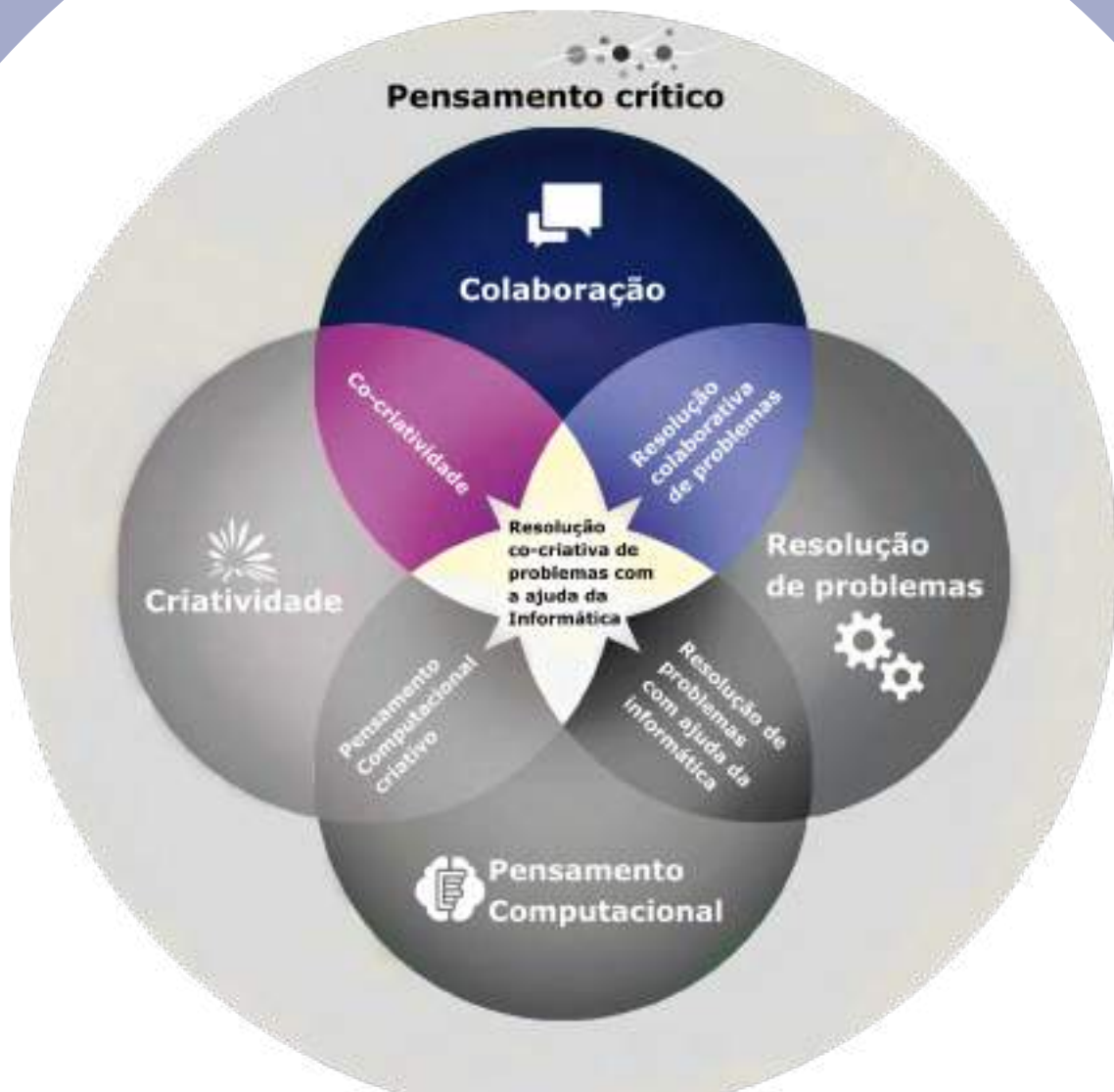
Componentes das Competências para o século 21:



O componente **pensamento crítico (CrT)** é a capacidade de desenvolver uma reflexão crítica independente. O pensamento crítico permite a análise de idéias, de conhecimentos e de processos relacionados a um sistema de valores e julgamentos próprios do indivíduo. É um pensamento responsável que se baseia em critérios, que é sensível ao contexto e a outras pessoas. Seguem seus componentes e atitudes:

- **Componente 1 (CrTc1):** Identificar os componentes de uma ideia ou obra.
- **Componente 2 (CrTc2):** Explorar as diferentes perspectivas e posições em relação a uma ideia ou obra.
- **Componente 3 (CrTc3):** Posicionar-se em relação a uma ideia ou obra.

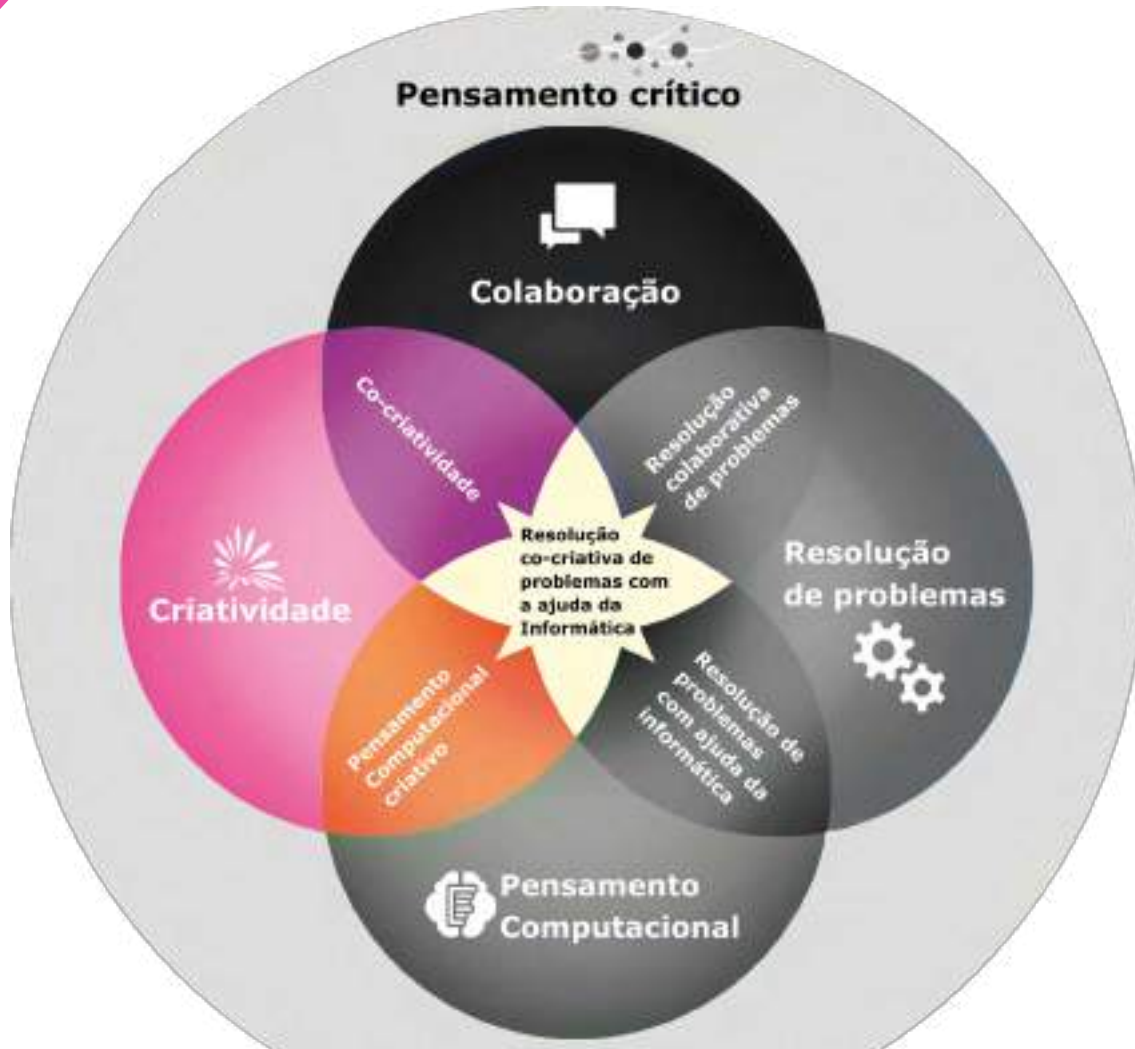




O **componente colaboração (C)** é a capacidade de desenvolver um entendimento compartilhado e trabalhar de forma coordenada com várias pessoas para um objetivo comum. Seguem seus componentes e atitudes:

- **Componente 1 (Cc1):** Capacidade de identificar a situação do problema e definir em equipe, um objetivo comum
- **Componente 2 (Cc2):** Estabelecer e manter um entendimento e uma organização compartilhada.
- **Componente 3 (Cc3):** Desenvolver uma compreensão do conhecimento, habilidades, pontos fortes e limitações de outros membros da equipe para organizar tarefas em direção a um objetivo comum.
- **Componente 4 (Cc4):** Ser capaz de gerenciar as dificuldades do trabalho em equipe com respeito e em busca de soluções.
- **Componente 5 (Cc5):** (Co)construção de conhecimento e / ou artefatos





O **componente criatividade (CR)** é um processo de criação de uma solução considerada nova, inovadora e relevante para abordar uma situação-problema e adaptada ao contexto. Seguem seus componentes, subcomponentes e atitudes:

➤ **Componente 1 (CRc1):** Incubação de ideias

- CRc1sc1: Explorar uma variedade de novas abordagens ou soluções
- CRc1sc2: Incubação de diferentes oportunidades, estratégias e conceitos

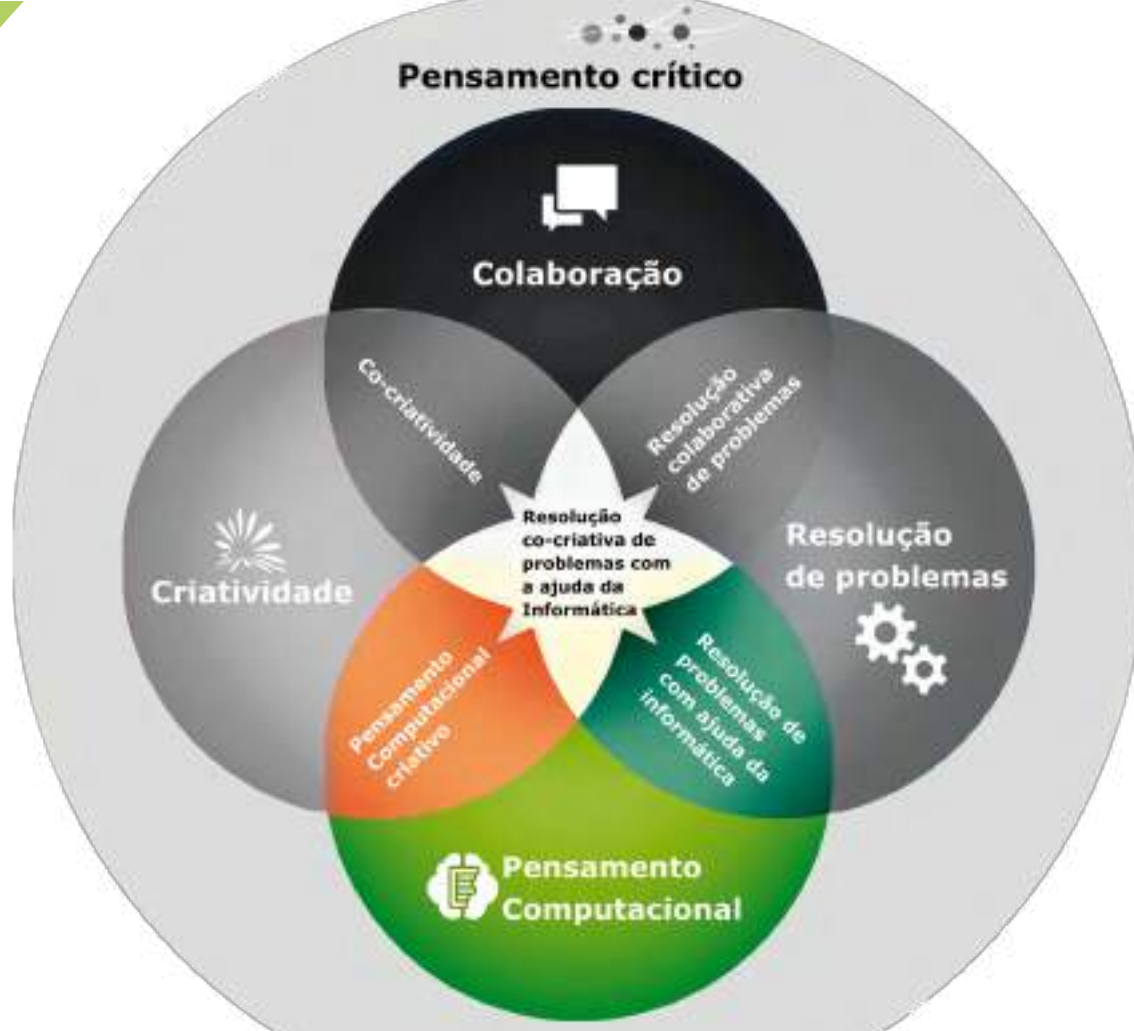
➤ **Componente 2 (CRc2):** Geração de idéias

- CRc2sc1: Geração de ideias, brainstorming, exploração divergente
- CRc2sc2: Usar fontes de inspiração para orientar a pesquisa criativa
- CRc2sc3: Combinação de ideias

➤ **Componente 3 (CRc3):** Avaliação e seleção

- CRc3sc1: Relevância e pertinência com relação à situação-problema
- CRc3sc2: Novidade e originalidade
- CRc3sc3: Elegância e parcimônia
- CRc3sc4: Capacidade de gerar novas perspectivas





O **componente pensamento computacional (CT)** é um conjunto de estratégias cognitivas e metacognitivas relacionadas à modelagem de conhecimento e de processos, à abstração, ao algoritmo, à identificação, à decomposição e à organização de estruturas complexas e conjuntos lógicos. Seguem seus componentes, subcomponentes e atitudes:

► **Componente 1 (CTc1): Análise (Entender uma situação e identificar componentes)**

- CTc1sc1: Compreender/identificar uma situação problema
- CTc1sc2: Escolher um ponto de vista/ângulo/perspectiva para analisar a situação (por exemplo, de outra pessoa, de uma máquina ou de um objeto)
- CTc1sc3: Identificar os principais objetos de uma situação, distinção de diferentes tipos de componentes (objetos, atributos, funções, eventos, padrões)
- CTc1sc4: Identificar os atributos de cada objeto e sua mudança ao longo do tempo (se dinâmico)
- CTc1sc5: Identificar as semelhanças ou as diferenças entre os objetos
- CTc1sc6: Identificar os eventos e o que eles acionam
- CTc1sc7: Identificar os relacionamentos entre os objetos ou entidades (por exemplo, dependência, hierarquia, causalidade)

➤ **Componente 2 (CTc2):** Modelagem (Capacidade de organizar e modelar uma situação)

- CTc2sc1: Reconhecer a importância de planejar uma solução antes de criá-la/implementá-la
- CTc2sc2: Escolher uma ferramenta adequada para definir um modelo de dados e seus relacionamentos
- CTc2sc3: Produzir uma representação/modelagem os dados de uma maneira que represente a situação
- CTc2sc4: Otimizar a organização dos dados (por exemplo, para evitar repetições)

➤ **Componente 3 (CTc3):** Alfabetização para codificar

- CTc3sc1: Escrever um algoritmo em uma linguagem natural ou pseudo-código
- CTc3sc2: Saber explicar o significado de um algoritmo existente
- CTc3sc3: Entender e identificar os conceitos de programação: blocos, objetos, instruções e operadores
- CTc3sc4: Fazer alterações em um algoritmo existente, ou criar um algoritmo a partir da análise e modelagem de uma situação ou com um modelo de dados existente
- CTc3sc5: Avaliar pedaços de código existente e propor otimizações e melhorias

➤ **Componente 4 (CTc4):** Alfabetização tecnológica e de Sistemas

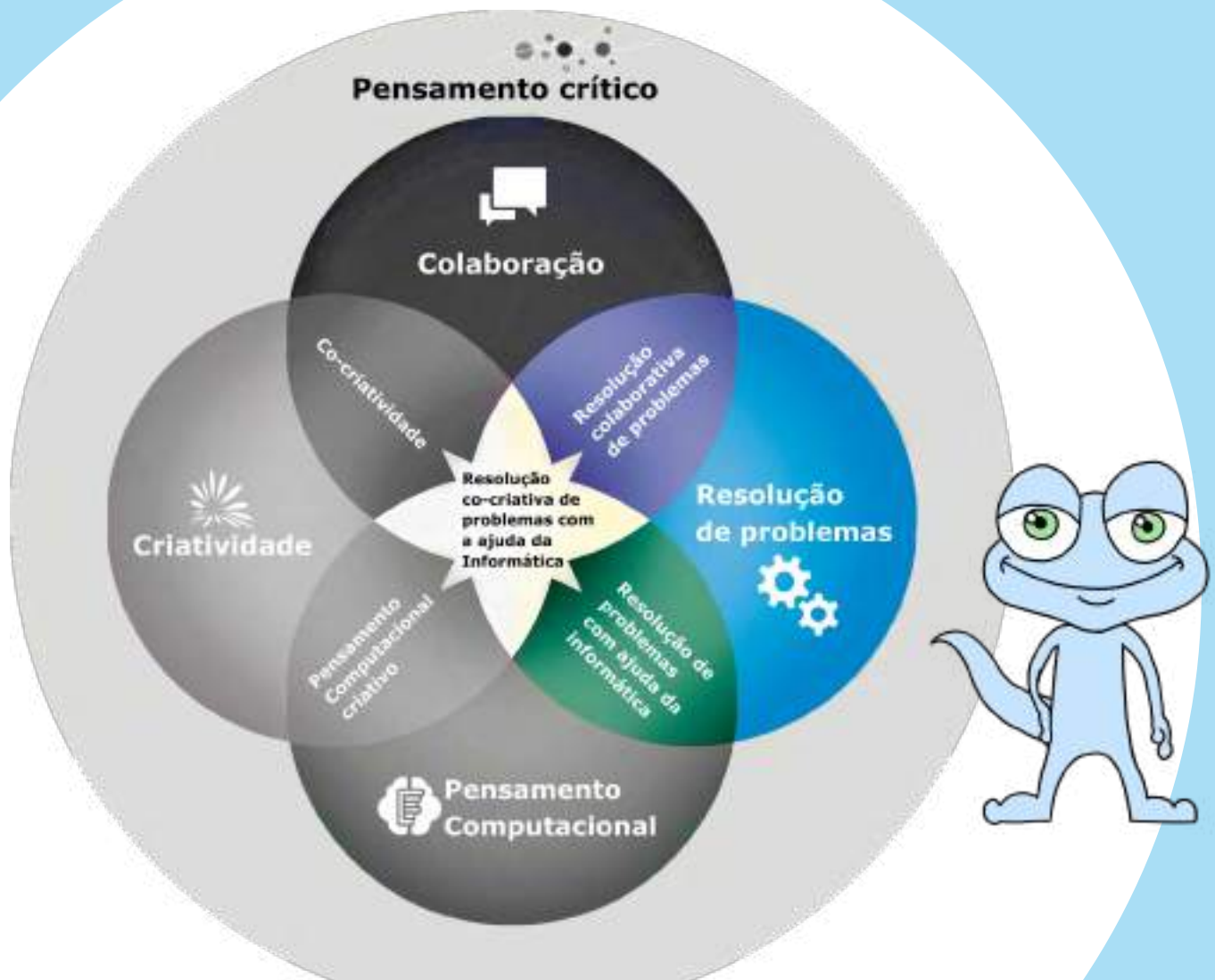
- CTc4sc1: Entender os recursos, funcionalidades e componentes de diferentes tecnologias (rede, computador, robô...)
- CTc4sc2: Saber escolher uma ou mais tecnologias adaptadas para a análise/modelagem da situação
- CTc4sc3: Entender as diferenças entre o software e o hardware
- CTc4sc4: Entender as abordagens tecnológicas de eletrônica, redes. Saber identificar os limites de um software ou de hardware
- CTc4sc5: Entender e descrever, de forma coerente, o funcionamento de um sistema complexo

➤ **Componente 5 (CTc5):** Programação

- CTc5sc1: Escolher uma linguagem apropriada a situação, ou uma tecnologia (robô, tablet..)
- CTc5sc2: Decompor um objetivo em uma sequência de instruções menores e que sejam suficientemente precisas para a linguagem de computador escolhida
- CTc5sc3: Identificar ou escrever funções ou blocos de código para um determinado objetivo
- CTc5sc4: Programar usando técnicas oferecidas pelo linguagem selecionada
- CTc5sc5: Entregar um programa livre de erros

➤ **Componente 6 (CTc6): Abordagem ágil e iterativa**

- CTc6sc1: Analisar se existem erros levando em consideração a melhoria do programa de computador
- CTc6sc2: Aproximando-se de uma solução com uma abordagem de resolver o problema por tentativa-erro
- CTc6sc3: Adotar uma abordagem iterativa baseada na otimização do programa



O **componente resolução de problemas (PS)** é a capacidade de identificar uma situação de problema, para a qual o processo e a solução não são conhecidos antecipadamente. É também a capacidade de determinar uma solução, construí-la e implementá-la efetivamente. Seguem seus componentes, subcomponentes e atitudes (por padrão, o conjunto de subcomponentes é considerado em tarefas colaborativas (isso pode dizer respeito a uma atividade individual ou colaborativa), o código * -ind indica subcomponentes existentes em tarefas individuais):

- **Componente 1 (PSc1):** Estabelecer e manter um entendimento compartilhado
 - PSc1sc1: Descobrir os pontos de vista e habilidades de outros membros da equipe
 - PSc1sc2: Construir uma representação compartilhada e negociar o significado do problema ou atividade a ser realizada (espaço compartilhado)
 - PSc1sc3: Comunicar-se com os membros da equipe sobre as ações a serem tomadas ou realizadas
 - PSc1sc4: "Monitorar" e realinhar o entendimento compartilhado
- **Componente 2 (PSc2):** Realizar ações apropriadas para resolver o problema
 - PSc2sc1: Descobrir o tipo de interação colaborativa para resolver o problema, bem como os objetivos
 - PSc2sc2: (ind) Identificar e descrever as tarefas a serem cumpridas/finalizadas
 - PSc2sc3: (ind) Implementar os planos/planejamentos
 - PSc2sc4: "Monitorar" os resultados das ações e avaliar o sucesso da resolução de problemas; (ind) Seguir o planejamento e regular a própria atividade
- **Componente 3 (PSc3):** Estabelecer e manter a organização da equipe
 - PSc3sc1: Entender os papéis/qualificações dos membros da equipe para que possam resolver o problema
 - PSc3sc2: Descrever os papéis e a organização da equipe (protocolo de comunicação / regras de engajamento)
 - PSc3sc3: Seguir as regras de engajamento (por exemplo, incentivar outros membros da equipe a concluírem suas tarefas.)
 - PSc3sc4: "Monitorar", fornecer feedback e adaptar a organização e os papéis da equipe
- **Componente 4 (PSc4):** Co-regulação iterativa de soluções intermediárias
 - PSc4sc1: (ind) Desenvolver confiança, aceitação de falhas e resiliência ao avaliar soluções intermediárias para o problema
 - PSc4sc2: (ind) Ser capaz de conceber a resolução de problemas como uma abordagem iterativa orientada para protótipos
 - PSc4sc3: Co-regular os esforços como iterações de soluções intermediárias que levam a uma solução ótima; (ind) Regular os esforços como iterações de soluções intermediárias que levam a uma solução ótima
 - PSc4sc4: (ind) Avaliar as vantagens e desvantagens de soluções intermediárias e adaptar futuras iterações de acordo
- **Componente 5 (PSc5):** Pesquisar e compartilhar recursos externos
 - PSc5sc1: Analisar se existem erros levando em consideração a melhoria do programa de computador
 - PSc5sc2: Aproximando-se de uma solução com uma abordagem de resolver o problema por tentativa-erro
 - PSc5sc3: Adotar uma abordagem iterativa baseada na otimização do programa
 - PSc5sc4: Implementar códigos de reinicialização para fazer o programa voltar ao estado inicial se necessário. Levar em conta o estado (inicial, atual ou final) do programa
 - PSc5sc5: Avaliar a relevância dos objetos representados/ operacionalizados em um programa em relação a uma situação problema

* COMO LER OS CÓDIGOS: por exemplo, o componente PSc1, significa o componente 1 da Resolução de Problemas, que possui subcomponentes, por exemplo, o código PSc5sc1, significa o subcomponente 1 do componente 5 da Resolução de Problemas.



Pistas para avaliação dos desafios segundo as Valores para os estudantes em um grupo de aprendizagem(#5c21)

Grupo
Boa vontade
Ajuda/assistência mútua
Equidade
Confiança
Interdependência
Cortesia e calma
Desenvolvimento sustentável

Individual
Hedonismo (ter prazer ao realizar as atividades)
Sucesso
Qualidade
Liberdade
Integridade
Privacidade
Usabilidade
Autonomia

Valores para os estudantes em uma comunidade de aprendizagem #5c21





Pistas para avaliação dos desafios segundo as Atitudes individuais relacionadas às Competências para o século 21 (#5c21)

Atitudes para o Pensamento Crítico (CrTat)

- CrTat1: Curiosidade
- CrTat2: Orientação para a aprendizagem
- CrTat3: Sensibilidade social e cultural
- CrTat4: Sensibilidade à diversidade
- CrTat5: Autonomia (ação e pensamento)

Atitudes para a Colaboração (Cat)

- Cat1: Senso de iniciativa
- Cat2: Flexibilidade/adaptabilidade
- Cat3: Liderança
- Cat4: Responsabilidade
- Cat5: Gestão de conflitos

Atitudes para a Resolução de Problemas (PSat)

- PSat1: Determinação, resiliência, perseverança
- PSat2: Aceitação de erros

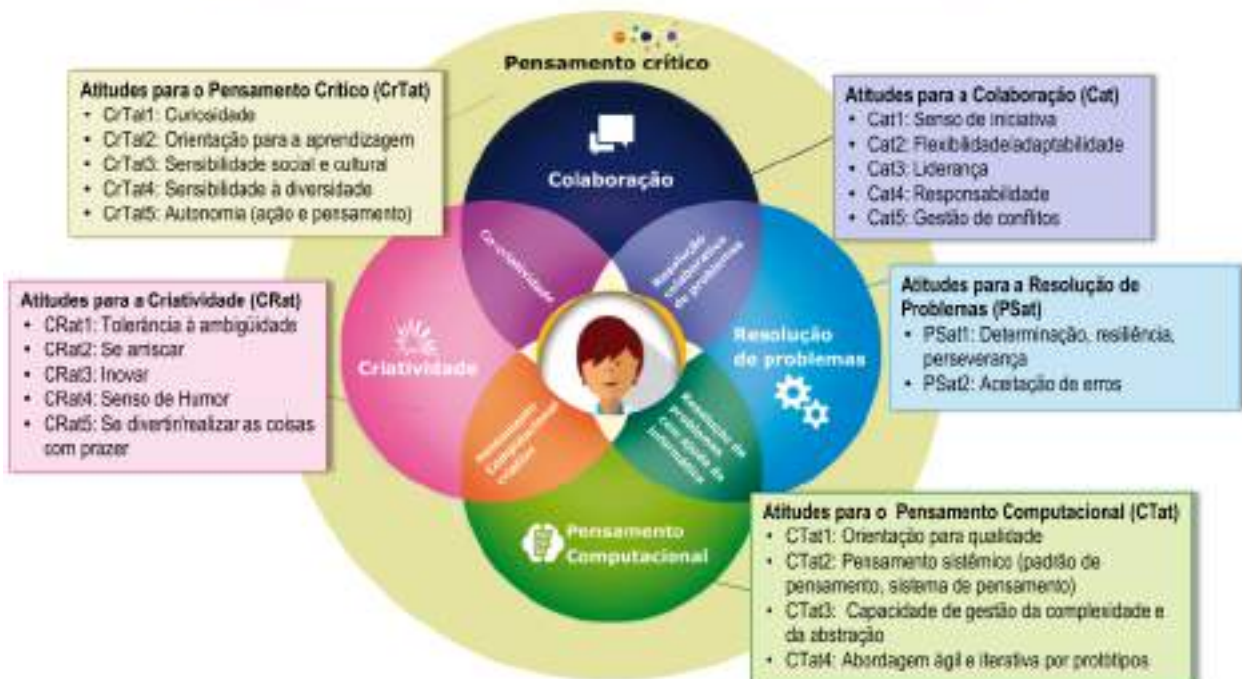
Atitudes para a Criatividade (CRat)

- CRat1: Tolerância à ambigüidade
- CRat2: Se arriscar
- CRat3: Inovar
- CRat4: Senso de Humor
- CRat5: Se divertir/realizar as coisas com prazer

Atitudes para o Pensamento Computacional (CTat)

- CTat1: Orientação para qualidade
- CTat2: Pensamento sistêmico (padrão de pensamento, sistema de pensamento)
- CTat3: Capacidade de gestão da complexidade e da abstração
- CTat4: Abordagem ágil e iterativa por protótipos

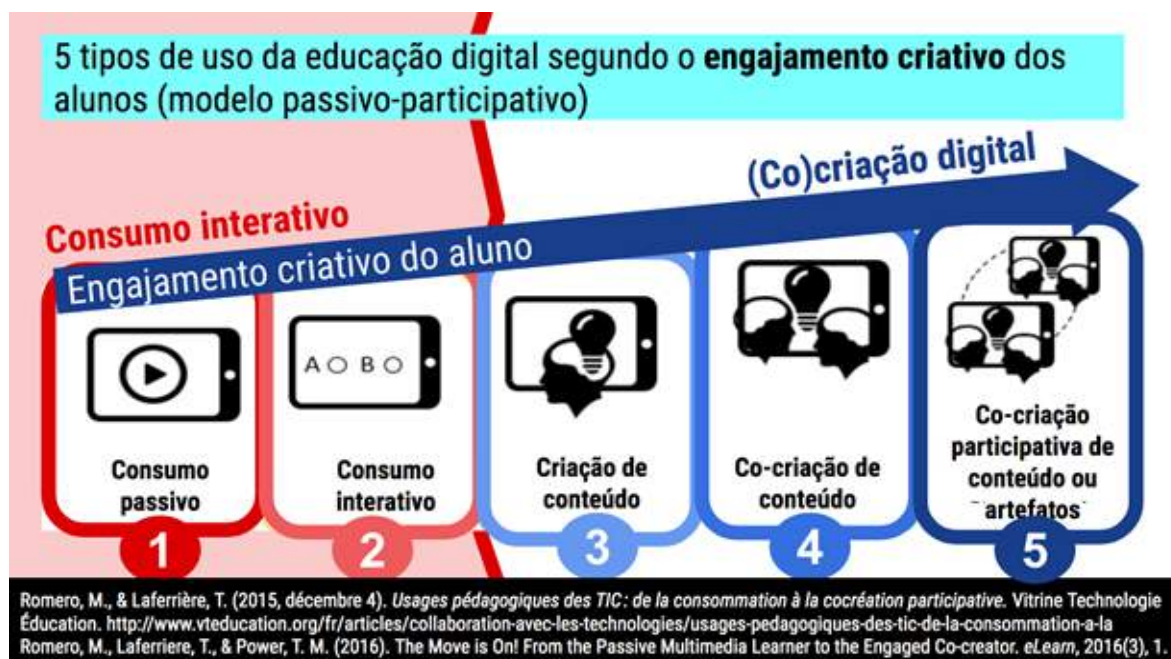
Atitudes individuais relacionadas às competências do século 21 #5c21





Pistas para avaliação dos desafios segundo o Engajamento Criativo (#5c21)

O engajamento criativo pode ser usado de forma digital ou de forma unplugged, pois o aluno pode desenvolver suas atividades de forma cinestésica. Veja mais dicas no artigo citado abaixo.



Formulário para AVALIAÇÃO do Pensamento Computacional & Competências para o século 21 (#5c21)

Auto-avaliação sobre Pensamento Computacional, segundo Yağcı (2019) :
<https://forms.gle/mapUHiwgVPKErn648>

Avaliação das Competências das crianças para o Século 21 (#5c21) nas atividades desenvolvidas por professores: <https://forms.gle/ZFBwQH57qeGgG5YT9>

Avaliação das Atitudes e Valores das crianças para o Século 21 (#5c21) nas atividades desenvolvidas por professores: <https://forms.gle/ZFBwQH57qeGgG5YT9>

BIBLIOGRAFIA

Adaptado da obra: Romero, M., Roy, A, Lepage, A. (2016). Défis de programmation créative: du conte au code avec Scratch et Vibot. Université Laval-Québec. Disponível em: <https://goo.gl/bBWGIk> (Publicação original).

Romero, M. et Vallerand, V. (2016). Guide d'activités technocréatives pour les enfants du 21e siècle. Université Laval-Québec. Disponível em: <https://lel.crires.ulaval.ca/oeuvre/guide-dactivites-technocreatives-pour-les-enfants-du-21e-siecle>

Romero, M.; Vallerand, V.; Nunes, M. A. S. N. (2019) Almanaque Para Popularização De Ciência Da Computação. Série 12: Guia Pedagógico; Volume 1: Atividades Tecnocriativas para crianças do século 21. ed. 1. Porto Alegre: SBC. v. 1. Disponível em: <http://almanaquesdacomputacao.com.br/gutanunes/publications/S12V1.pdf>

Romero, M.; Insfrán, D. A.; Nunes, M. A. S. N.; Loufane, S.F. (2019) Almanaque Para Popularização De Ciência Da Computação. Série 12: Guia Pedagógico; Volume 5: Betabot - a representante brasileira da Liga dos Bots para o desenvolvimento do pensamento computacional no Brasil. ed. 1. Porto Alegre: SBC. v. 5. Disponível em: <http://almanaquesdacomputacao.com.br/gutanunes/publications/S12V5.pdf>

Romero, M., Lepage, A., & Lille, B. (2017). Computational thinking development through creative programming in higher education. International Journal of Educational Technology in Higher Education, 14(1), 42.

Romero, M. et al. (2017). Assessment of creativity and computational thinking in Scratch projects. In: Scratch conference. #Scratch2017BDX. Disponível em: <https://vibot.fse.ulaval.ca/2017/07/19/scratch2017bdx-assessment-of-creativity-and-computational-thinking-in-scratch-projects/>

Romero, M. (2016). De l'apprentissage procédural de la programmation à l'intégration interdisciplinaire de la programmation créative. Formation et profession, 24(1), 87-89. <https://doi.org/10.18162/fp.2016.a92>

Romero, M., Laferriere, T., & Power, T. M. (2016). The Move is On! From the Passive Multimedia Learner to the Engaged Co-creator. eLearn, 2016(3), 1.

Romero, M., Davidson, A-L., Cucinelli, G., Ouellet, H., & Arthur, K. (2016). Learning to code: from procedural puzzle-based games to creative programming. CIDUI.

Romero, M., & Laferrière, T. (2015, décembre 4). Usages pédagogiques des TIC : de la consommation à la cocréation participative. Vitrine Technologie Éducation. <http://www.vteducation.org/fr/articles/collaboration-avec-les-technologies/usages-pedagogiques-des-tic-de-la-consommation-a-la>

Romero, M., Laferriere, T., & Power, T. M. (2016). The Move is On! From the Passive Multimedia Learner to the Engaged Co-creator. eLearn, 2016(3), 1.

Romero, M. e Loufane, S.F. Vibot the Robot: Introduction to programming. Université Laval/ Québec. Ed. Publications Québec. p.24. 2016. Disponível em <https://www.amazon.fr/>

Yağcı, M. A valid and reliable tool for examining computational thinking skills. IN: Educ Inf Technol (2019). 24: 929. <https://doi.org/10.1007/s10639-018-9801-8>

Mais gibis, cartilhas e guias em: <http://almanaquesdacomputacao.com.br/>

AUTORES



Margarida Romero

É Diretora de pesquisa do Laboratoire d'Innovation et Numérique pour l'Education (LINE), um laboratório na área de Technology Enhanced Learning (TEL). Ela é professora efetiva a Université Cotê d'Azur (France) e professora associada a Université Laval no (Canadá). Sua pesquisa é orientada para os usos

inclusivos, humanísticos e criativos das tecnologias (co-design, game design e robótica) para o desenvolvimento da criatividade, resolução de problemas, colaboração e pensamento computacional. Ela é responsável pela concepção filosófica, planejamento e criação da versão conceitual do Vibot. <https://margaridaromero.wordpress.com/>

Linkedin: <https://www.linkedin.com/in/margarida/>



Maria Augusta Silveira Netto Nunes

Bolsista de Produtividade Desen. Tec. e Extensão Inovadora do CNPq - Nível 2 - CA 96 - Programa de Desenvolvimento Tecnológico e Industrial

É Professor Associado II do Departamento de Computação da Universidade Federal de Sergipe. Membro do Programa de Pós-graduação em

Ciência da Computação (PROCC) na UFS. Pós-doutora pelo laboratório LINE, Université Côte d'Azur/Nice Sophia Antipolis/ Nice-França (2019). Pós-doutora pelo Instituto Nacional de Propriedade Industrial (INPI) (2016). Doutora em "Informatique pela Université de Montpellier II - LIRMM em Montpellier, França (2008). Realizou estágio doutoral (doc-sanduíche) no INESC-ID- IST Lisboa- Portugal (ago 2007-fev 2008). Mestre em Ciência da Computação pela Universidade Federal do Rio Grande do Sul (1998). Graduada

pela Universidade Federal do Rio Grande do Sul (1998). Graduada em Ciência da Computação pela Universidade de Passo Fundo-RS (1995). É bolsista produtividade DT-CNPq. Atualmente, suas pesquisas estão voltadas principalmente na área de Pensamento Computacional usando gibis/ HQs. Criou o projeto "Almanaques para Popularização de Ciência da Computação" chancelado pela SBC.

Lattes: <http://lattes.cnpq.br/9923270028346687>



Luís Antônio dos Santos Silva

É técnico formado pelo Instituto Federal de Sergipe (IFS). Graduado pela Universidade Tiradentes (UNIT) Bacharel em Sistemas de Informação atualmente discente do Programa de Pós-Graduação em Ciências da Computação (PROCC) da Universidade Federal de Sergipe (UFS). Atualmente, suas pesquisas estão voltadas principalmente na área de Pensamento Computacional.

Lattes: <http://lattes.cnpq.br/1622211414785978>



José Humberto dos Santos Júnior

Bolsista CNPq - Iniciação Tecnológica

Estudante de Ciência da Computação da Universidade Federal de Sergipe – UFS.

Lattes: <http://lattes.cnpq.br/9144803555676838>



Aurélie Roy

É estudante de Literatura Inglesa na Université de Laval- Quebec/CA. Seu forte interesse em idiomas, educação e criação deu a ela a oportunidade de se juntar à equipe #CoCreaTIC em 2016. Ela desenvolveu recentemente uma paixão por programação criativa. Ela trabalha com Scratch. Em breve inicia seu doutorado em Literatura Inglesa



Alexandre Lepage

Alexandre Lepage é professor do ensino médio. Ele está cursando mestrado em Tecnologia Educacional na Université de Laval- Quebec/CA. Tem interesse em aprender programação para desenvolver estratégias metacognitiva e aplicá-las ao ensino.

Linkedin: <https://www.linkedin.com/in/alexandre-lepage-8b395351/>

Agradecimentos

Ao CNPq, CAPES, SBC, BICEN, DCOMP/PROCC, UNIRIO/UNIRIOTEC, Université Laval, Publications Québec, Université Nice Sophia Antipolis, Université Côte d'Azur, LINE- Laboratoire.

APOIO:



Membre de UNIVERSITÉ CÔTE D'AZUR



Laboratoire d'Innovation et Numérique pour l'Éducation



UNIVERSIDADE
FEDERAL DE
SERGIPE



Conselho Nacional de Desenvolvimento
Científico e Tecnológico



UNIVERSITÉ
LAVAL



ISBN 978-857669482-3



9

788576

694823