

Produto Educacional

INTRODUÇÃO AO PENSAMENTO COMPUTACIONAL COM O USO DA PLATAFORMA GENIA



Letícia Carla de Carvalho
Marco Aurélio Kalinke
2025



TERMO DE LICENCIAMENTO

ESTA DISSERTAÇÃO E O SEU RESPECTIVO PRODUTO EDUCACIONAL ESTÃO LICENCIADOS SOB CREATIVE COMMONS ATRIBUIÇÃO-NÃOCOMERCIAL-COMPARTILHAMENTO PELA MESMA LICENÇA 4.0 INTERNACIONAL. PARA VISUALIZAR UMA CÓPIA DESTA LICENÇA, VISITE [HTTPS://CREATIVECOMMONS.ORG/LICENSES/BY-NC-](https://creativecommons.org/licenses/by-nc-sa/4.0/)

[SA/4.0/](https://creativecommons.org/licenses/by-nc-sa/4.0/)



Esta licença permite que outros remixem, adaptem e criem a partir do trabalho para fins não comerciais, desde que atribuam o devido crédito e que licenciem as novas criações sob termos idênticos. Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença



Apresentação

Prezados professores,

Este material apresenta o produto educacional resultante da pesquisa de mestrado intitulada "Possibilidades de uso da Plataforma GenIA para a Construção de Algoritmos: Um Auxílio para o Desenvolvimento do Pensamento Computacional", realizada no Programa de Pós-Graduação em Formação Científica, Educacional e Tecnológica (PPGFCET) da Universidade Tecnológica Federal do Paraná (UTFPR), sob orientação do Prof. Dr. Marco Aurélio Kalinke.

O produto é este ebook denominado "Introdução ao Pensamento Computacional com o Uso da Plataforma GenIA". Ele foi elaborado com o objetivo de apoiar o planejamento de aulas do componente curricular de Pensamento Computacional, promovendo o uso da plataforma GenIA tanto em atividades de laboratório quanto em propostas desenvolvidas em sala de aula. A intenção é favorecer a construção dos nexos conceituais indicados por Navarro (2021) e o desenvolvimento dos pilares do Pensamento Computacional conforme propostos por Brackmann (2017).

Por meio da pesquisa realizada, constatamos que a programação por fluxograma, mediada pela GenIA, é uma abordagem promissora para o ensino de algoritmos e conceitos matemáticos, especialmente por tornar mais visível e compreensível o processo passo a passo envolvido nessas áreas. Contudo, a autonomia dos estudantes ainda depende de intervenções pedagógicas consistentes e de melhorias na estabilidade e acessibilidade da plataforma.

Esperamos que este material possa ser apreciado, possibilitando entendimentos sobre a GenIA e o Pensamento Computacional e, conseqüentemente, contribuindo para a formação docente.



Atenciosamente,
Leticia Carla de Carvalho
Marco Aurélio Kalinke

01

Introdução ao Pensamento
Computacional.....Pg. 03

02

A plataforma GenIA.....Pg. 12

03

Planejamento de aulas
PLUGADAS.....Pg. 34

04

Planejamento de aulas
DESPLUGADAS.....Pg. 45

05

Referencial teórico.....Pg. 62

01

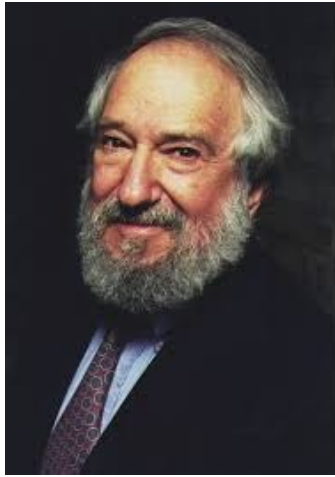
INTRODUÇÃO AO PENSAMENTO COMPUTACIONAL



AFINAL DE CONTAS, O QUE É PENSAMENTO COMPUTACIONAL?



As primeiras concepções sobre Pensamento Computacional (PC) foram desenvolvidas pelo matemático Seymour Papert, que defendia a ideia de que crianças podem aprender de forma significativa por meio da programação. Nessa perspectiva, ele contribuiu para a criação da linguagem Logo, desenvolvida no MIT em colaboração com Cynthia Solomon e Wally Feurzeig. O objetivo era possibilitar que os estudantes explorassem conceitos matemáticos e lógicos de maneira interativa, intuitiva e envolvente.



SEYMOUR PAPERT



CYNTHIA SOLOMON



WALLY FEURZEIG

Mais tarde, o termo ganhou maior visibilidade com o artigo publicado por Jeannette Wing, em 2006, intitulado *Computational Thinking*. Wing (2006) acreditava que o PC deveria ser essencial para todos, mesmo que não seja um profissional para a área da computação



JEANNETTE WING

Com a visão voltada para a ciência da computação, em 2011 Wing definiu o PC como sendo “os processos de pensamento envolvidos na formulação de problemas e suas soluções, de modo que estas possam ser representadas de maneira a serem efetivamente executadas por um agente de processamento de informações” (WING, 2011, p. 20, tradução nossa).

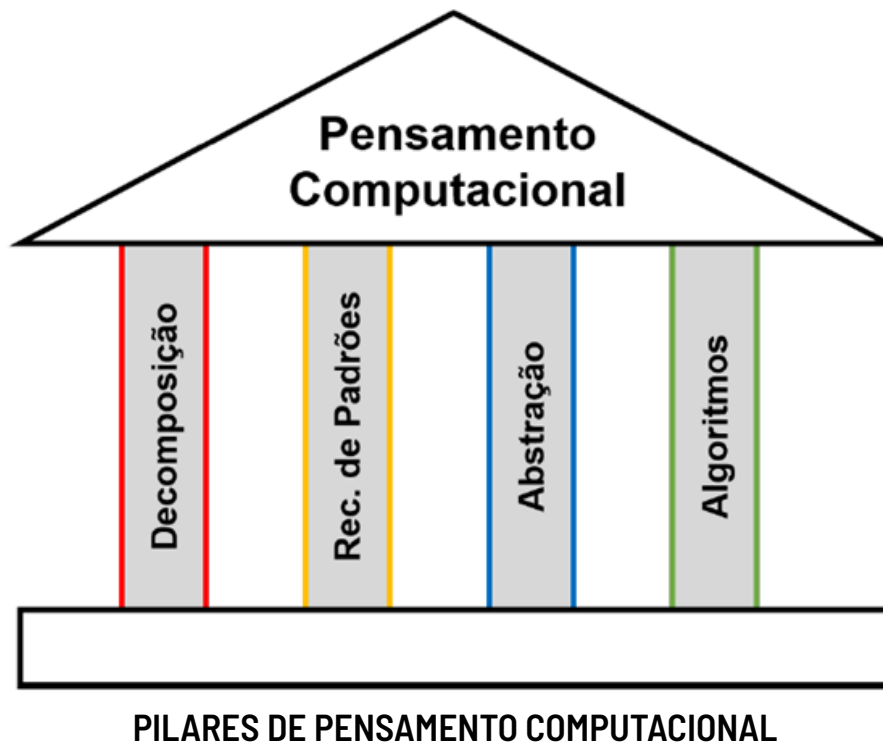


TARTARUGA LOGO DE PAPERT, SOLOMON E FEURZEIG



CHRISTIAN BRACKMANN

Ainda baseada nessa mesma visão, Christian Brackmann afirma que o PC é estruturado com base em quatro pilares fundamentais: decomposição, reconhecimento de padrões, abstração e algoritmos.



Para o autor:

“

O Pensamento Computacional envolve identificar um problema complexo e quebrá-lo em pedaços menores e mais fáceis de gerenciar (DECOMPOSIÇÃO). Cada um desses problemas menores pode ser analisado individualmente com maior profundidade, identificando problemas parecidos que já foram solucionados anteriormente (RECONHECIMENTO DE PADRÕES), focando apenas nos detalhes que são importantes, enquanto informações irrelevantes são ignoradas (ABSTRAÇÃO). Por último, passos ou regras simples podem ser criados para resolver cada um dos subproblemas encontrados (ALGORITMOS) (Brackmann, 2017, p. 33).

”

$$X = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad ax^2 + bx + c = 0$$

Ao se distanciar da abordagem da ciência da computação e considerar a presença do PC na BNCC inserido na área de matemática, Navarro (2021) propõe uma definição fundamentada em uma perspectiva nessa área. Para a autora, o PC é constituído por três nexos conceituais interligados: o pensamento algébrico, o pensamento algorítmico e a resolução de problemas, enfatizando a importância do desenvolvimento integrado desses elementos (Navarro, 2021, p. 134).



ELOISA NAVARRO



Este produto educacional foi desenvolvido com base nos pilares de Brackmann e nos nexos conceituais propostos por Navarro, orientando a elaboração de atividades plugadas e desplugadas.

NEXOS CONCEITUAIS:

Pensamento Algébrico

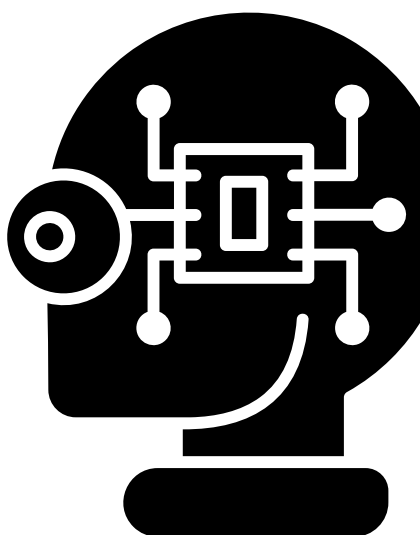
Navarro (2021, p. 118) conceitua o pensamento algébrico pela capacidade de identificar regularidades entre elementos de forma abstrata e generalizada, o que permite organizar a realidade. Ele é visto como uma atividade humana essencial para a criação de significados, servindo como uma ferramenta para sistematizar a realidade, desenvolver modelos e moldar diferentes contextos. Além disso, esse tipo de pensamento contribui para a formação e compreensão de conceitos, a organização do ambiente sociocultural, e a mediação entre as pessoas e o conhecimento.



NEXOS CONCEITUAIS:

Pensamento algorítmico

Navarro (2021, p. 128) compreende o pensamento algorítmico como um método aplicado à decomposição, resolução e execução de ações de forma ordenada, visando à obtenção de resultados baseados em padrões ou sequências lógicas. Esse tipo de pensamento envolve raciocínio matemático estruturado, uso de signos, operações, regras e etapas organizadas, sempre com foco na identificação de regularidades e na busca por generalizações. No âmbito educacional, seu desenvolvimento deve ocorrer por meio de atividades significativas, que incentivem os alunos a construir estratégias, elaborar hipóteses e resolver problemas, valorizando a intuição, a dedução e o raciocínio por analogia, em contraste com abordagens centradas na memorização. Destaca-se, ainda, que o pensamento algorítmico vai além da linguagem de programação, configurando-se como um processo lógico aplicável a diversas situações do cotidiano e à organização de contextos sociais complexos.



NEXOS CONCEITUAIS:

Resolução de Problemas

A resolução de problemas estabelece uma relação dialética entre o conhecimento já adquirido e o que precisa ser mobilizado, considerando aspectos cognitivos, afetivos e socioculturais. Na Educação Matemática, sob a Teoria Histórico-Cultural, Navarro (2021, p. 113) a descreve como um meio de desenvolver o pensamento matemático em situações concretas e significativas, aproximando a matemática do cotidiano. Orientada pedagogicamente pelo professor, envolve o uso de conceitos, raciocínio lógico, interpretação, planejamento e verificação de estratégias, superando a visão de um conhecimento puramente reprodutivo.

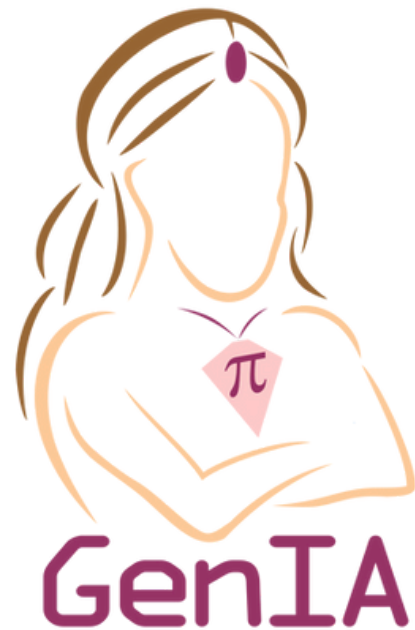


02

A PLATAFORMA GENIA



UMA NOVA FORMA DE PROGRAMAR: PLATAFORMA GENIA



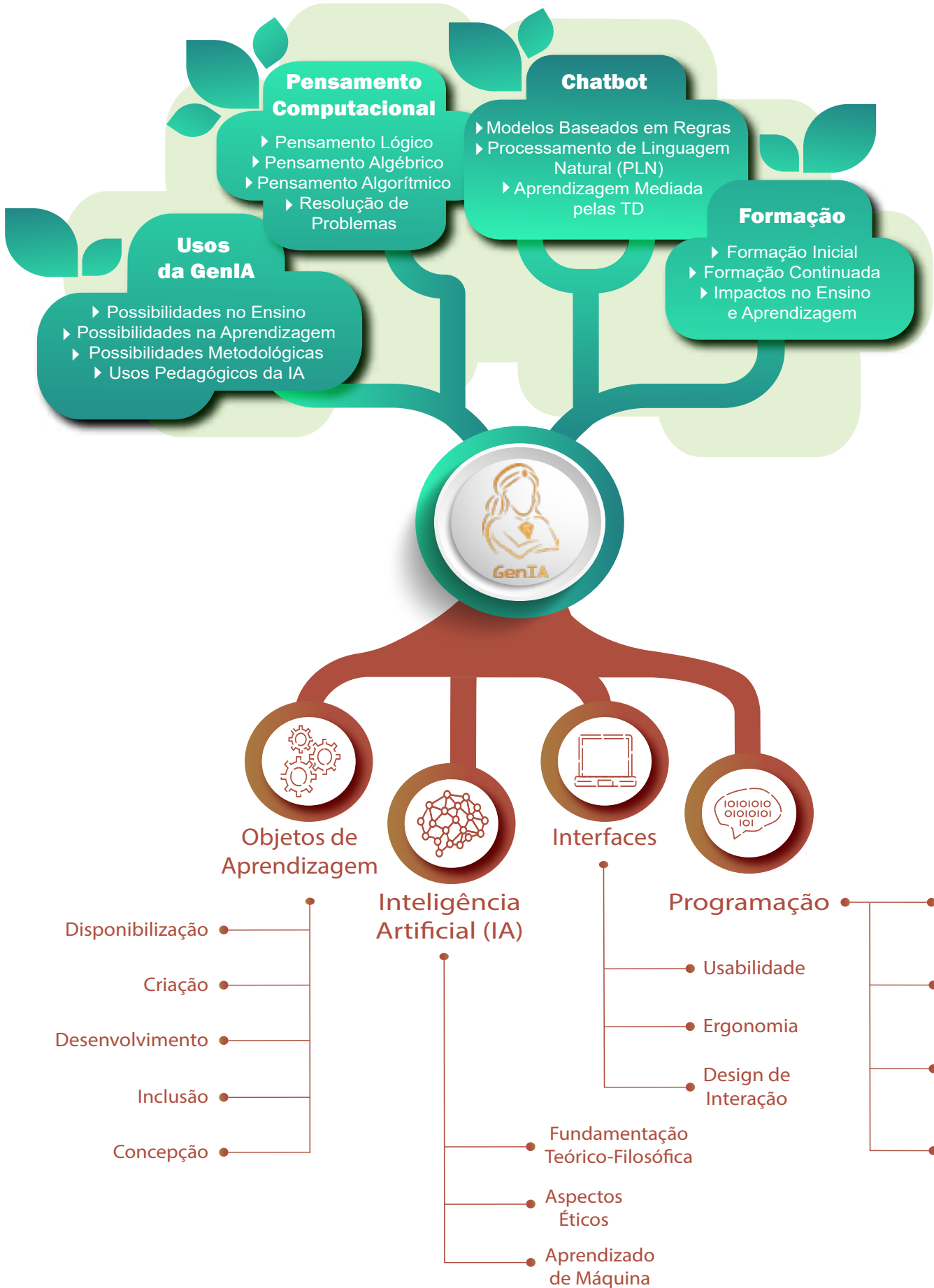
Os estudos sobre Objetos de Aprendizagem (OA) e Inteligência Artificial (IA) constituem temas centrais no Grupo de Pesquisa sobre Tecnologias na Educação Matemática (GPTEM). Entre as diversas discussões sobre o assunto, Zatti (2023, p. 18) destaca a proposta do professor Dr. Marco Aurélio Kalinke, líder do grupo e orientador, que idealizou um projeto de pesquisa voltado à criação de uma plataforma para a produção de OA assistido por IA, chamado de macroprojeto.

A GenIA integra esse macroprojeto, que será apresentado na próxima página. Para explorar cada uma das pesquisas vinculadas ao macroprojeto, basta clicar nas respectivas áreas de interesse.



EVANDRO ZATTI

Pesquisas macroprojeto GenIA



DOWNLOAD E INSTALAÇÃO DA PLATAFORMA GENIA



[Clique aqui para assistir o tutorial.](#)

Para fazer download da plataforma GenIA basta acessar o site: www.plataformagenia.com e acessar a aba download.



[Clique aqui para acessar o site](#)



E clicar no botão baixar:



Será feito o download de um arquivo no formato .zip. Após isso, basta extraí-lo e executar o instalador GenIA_Setup (X_X_X).msi, seguindo as instruções exibidas na tela.

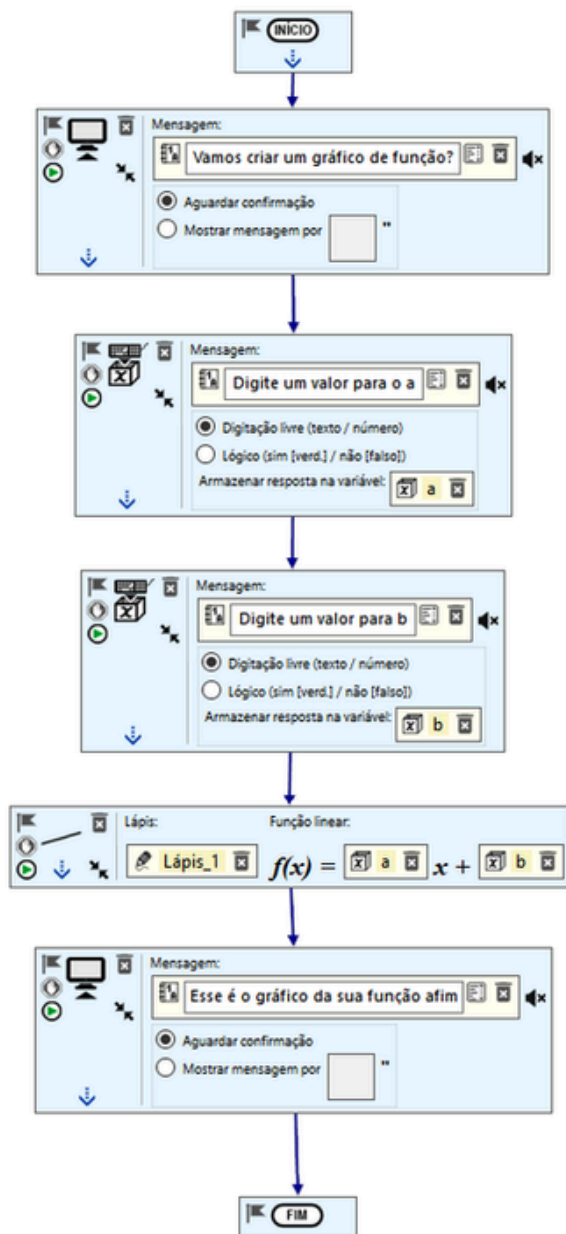
Atenção: Para que a GenIA funcione corretamente, é necessário instalar o .NET Desktop Runtime 8.x.

Caso o framework não esteja presente, a própria GenIA emitirá um aviso e redirecionará automaticamente para a página de download no site oficial:

<https://dotnet.microsoft.com/pt-br/download/dotnet/8.0>

A PLATAFORMA GENIA

A plataforma GenIA possui programação intuitiva baseada em fluxograma.



Programação intuitiva

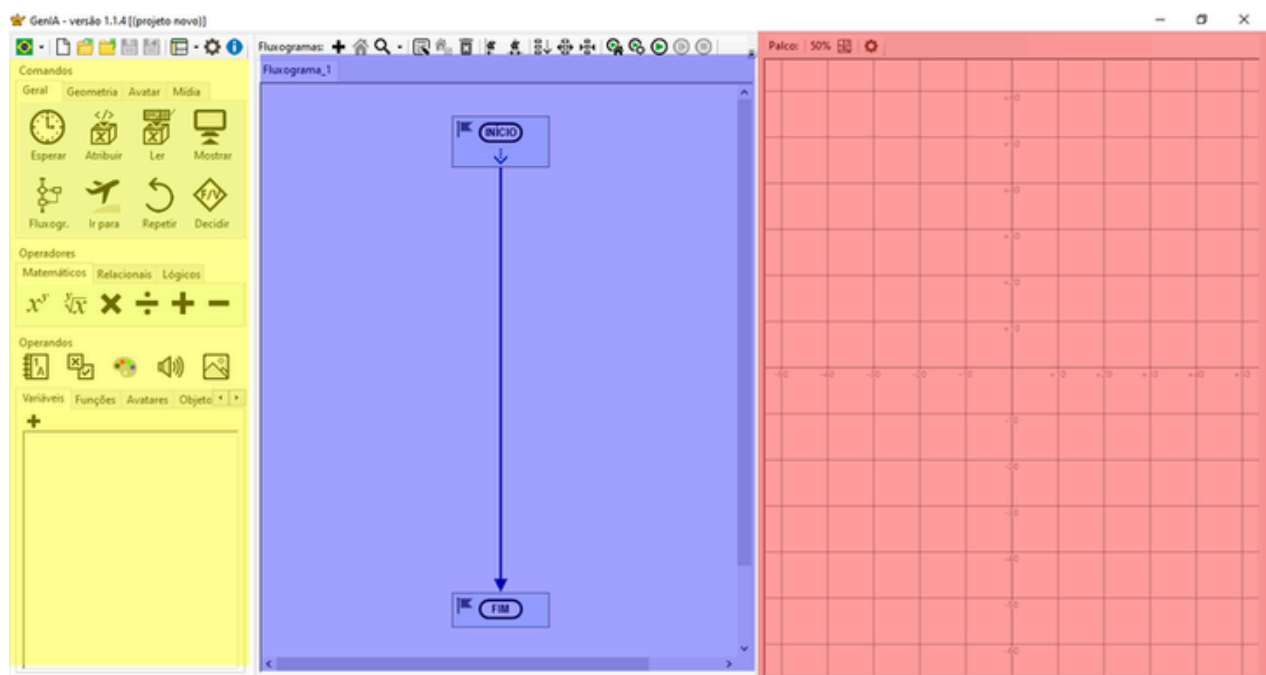
Uma linguagem de programação destinada à construção de projetos educacionais em ambientes computacionais que não necessitem o domínio de uma linguagem de programação específica e que apresentem características de similaridade, visualização e acessibilidade (Balbino et al., 2021, p. 19).

A notação gráfica por meio de fluxogramas utiliza símbolos e figuras para representar, de forma visual, o fluxo de execução de um processo. O fluxograma, nesse sentido, é uma representação gráfica de um algoritmo.

TELA DA PLATAFORMA

A interface da GenIA se divide em três áreas principais, sendo elas: Comandos, operadores e operandos (área amarela), área de programação do fluxograma (área azul) e palco (área vermelha).

ATENÇÃO: A presente publicação foi desenvolvida com base na versão 1.2.0 da GenIA.



A área amarela reúne os comandos com os blocos programáveis; já a área de operadores agrupa funções aritméticas, relacionais, lógicas e outras, incluindo operações com textos, áudios e imagens. A área de operandos, por sua vez, disponibiliza os elementos que podem ser combinados com esses operadores. A construção do algoritmo acontece na área azul do fluxograma, onde os comandos são organizados de forma visual. Por fim, a área vermelha do palco exibe o resultado da programação, permitindo o posicionamento dos elementos gráficos e sua visualização em tempo de execução, com opções de exibição em tela cheia ou embutida.

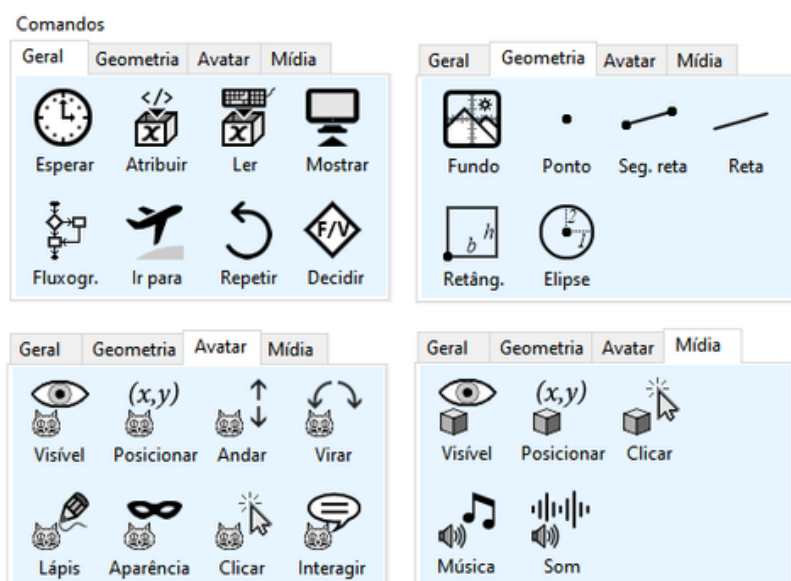
ÁREA DE COMANDOS

A área de comandos é organizada em diferentes abas, cada uma com funções específicas. A primeira aba "Geral" oferece comandos gerais, como ler e atribuir dados em variáveis, mostrar resultado de operações, esperar para executar uma próxima ação, alterar o fluxograma ou ir para um determinado ponto do fluxograma, repetir uma sequência de comandos, e realizar uma tomada de decisão, bastante conhecida no mundo da programação como se/senão ou if/else.

A aba "Geometria" oferece comandos para a construção de elementos geométricos básicos, como pontos, segmentos de reta, retas, retângulos e elipses. Além disso, o usuário pode alterar o fundo do OA criado.

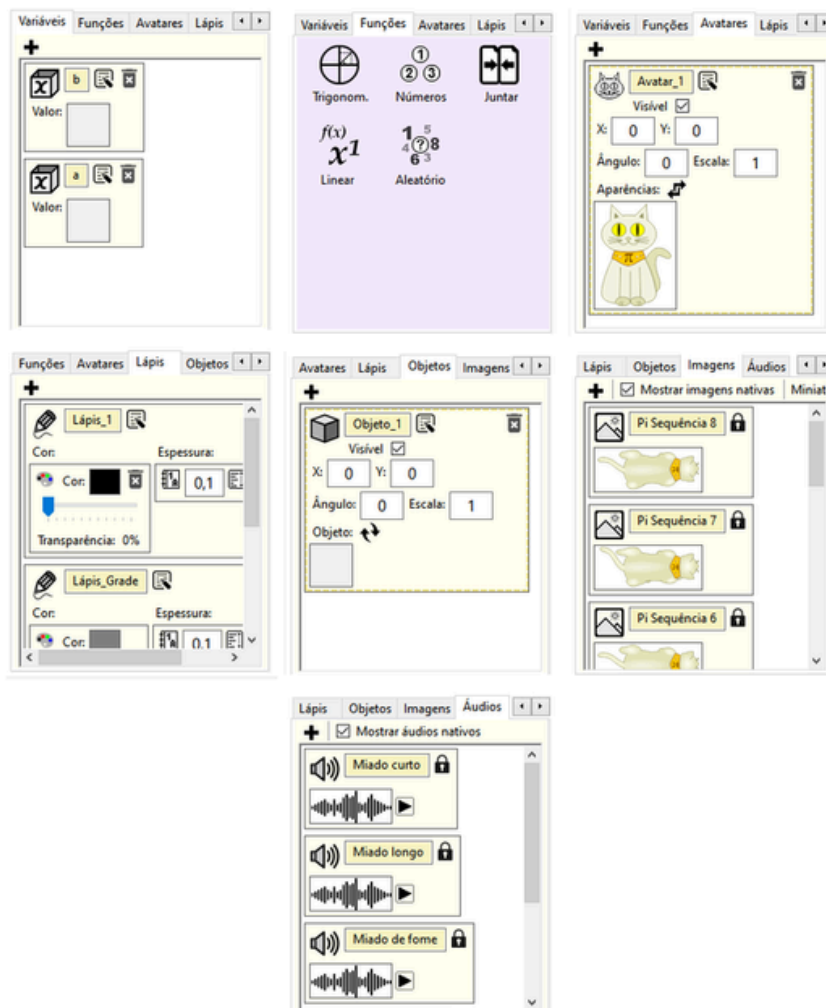
Na aba "Avatar", o usuário pode controlar o avatar escolhido, definindo sua visibilidade (visível ou invisível), posicionando-o em um ponto específico do palco (através de um plano cartesiano), fazendo-o andar, virar, mudar a aparência, executar ações ao ser clicado, interagir com falas ao usuário e escolher a cor da linha (lápiz) para desenhar objetos durante a programação.

Por fim, na aba mídia, temos as interações com sons que podemos incluir na programação, podendo ser música ou uma fala gravada.



OUTRAS ÁREAS DA GENIA

Além da área de operadores e operandos, a GenIA oferece um conjunto de abas adicionais que expandem as possibilidades de criação de OA, incluindo a aba "Variáveis", para criar variáveis, a aba "Funções", com ferramentas para a criação de funções como trigonométricas, operações com números, concatenação de textos (juntar), função linear e geração de números aleatórios, a aba "Avatares", para fazer upload de um avatar ou escolher um avatar existente na GenIA conhecido como Pi, um gatinho mascote da plataforma, as abas "Objetos", "Áudios" e "Imagens", para adicionar elementos multimídia ao programa criado, e finalizando com a aba "Lápis", que permite escolher as cores e espessuras das linhas dos objetos geométricos ou gráficos gerados no OA.

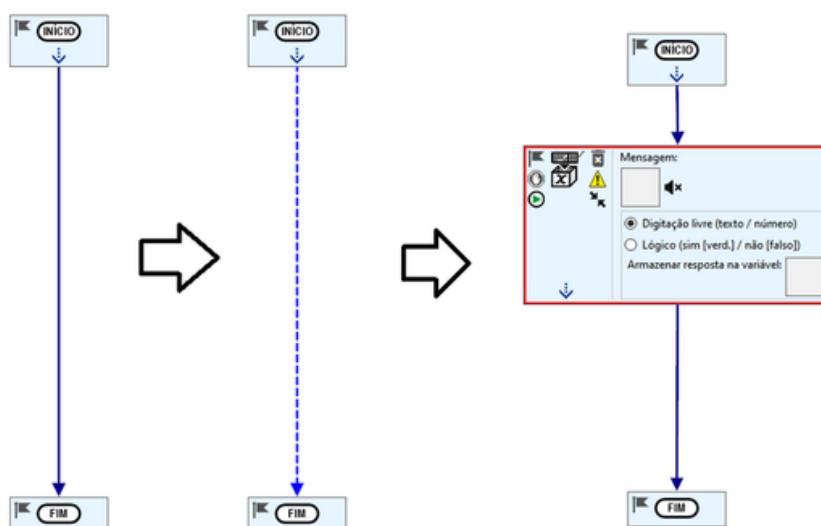


COMO REALIZAR A PROGRAMAÇÃO



[Clique aqui para assistir o tutorial.](#)

Para programar na GenIA, o usuário deve arrastar os comandos desejados até que a linha central da área de fluxograma fique pontilhada, indicando o local correto de inserção. Esse traçado serve como guia visual para o encaixe. Uma vez posicionado, basta soltar o comando para que ele seja integrado à programação.



Ao soltar o comando, ele será contornado em vermelho, sinalizando que ainda há informações obrigatórias a serem preenchidas para que a programação funcione corretamente. Quando todos os dados forem inseridos de forma adequada, o contorno vermelho desaparece, permitindo que o usuário avance para a próxima etapa.

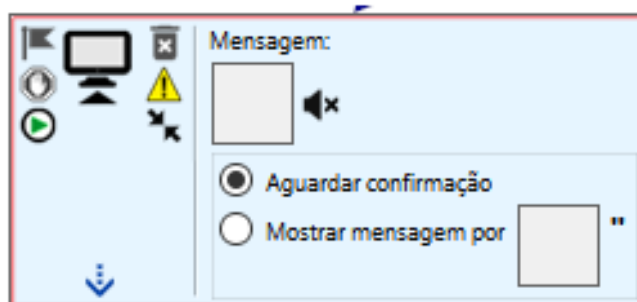
Observe que o fluxograma segue uma sequência lógica: cada comando leva ao próximo passo, ou seja, à próxima ação que se deseja que o programa execute.

GESTÃO DE ERRO NA GENIA

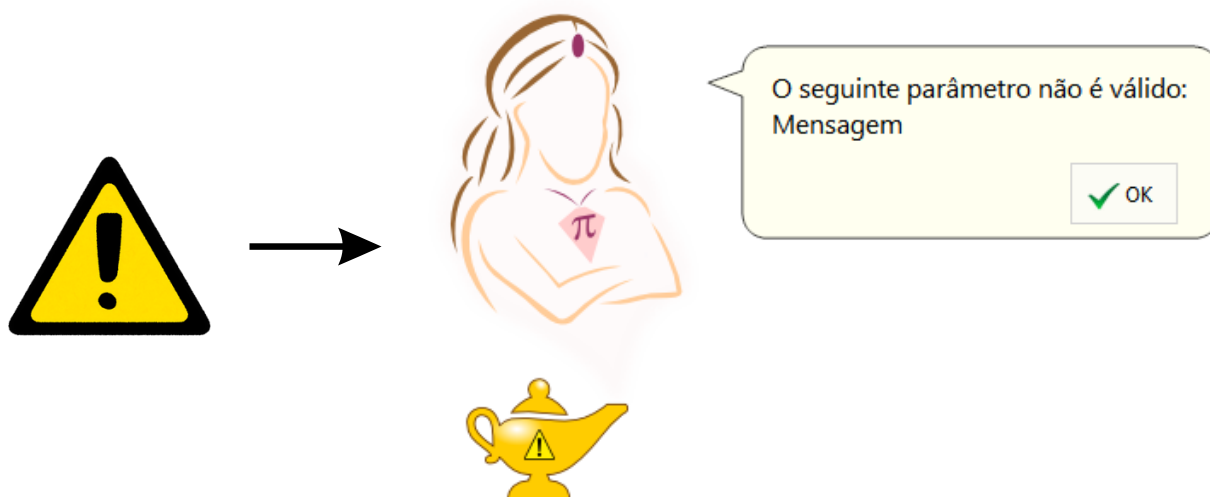


[Clique aqui para assistir o tutorial.](#)

Como mencionado, sempre que um comando estiver com informações incompletas para que a programação funcione corretamente, ele será destacado com uma borda vermelha.



Em caso de dúvida sobre as informações necessárias, basta clicar no triângulo com o ícone de exclamação; a GenIA fornecerá orientações sobre o que precisa ser preenchido.



CRIANDO VARIÁVEIS



[Clique aqui para assistir o tutorial.](#)

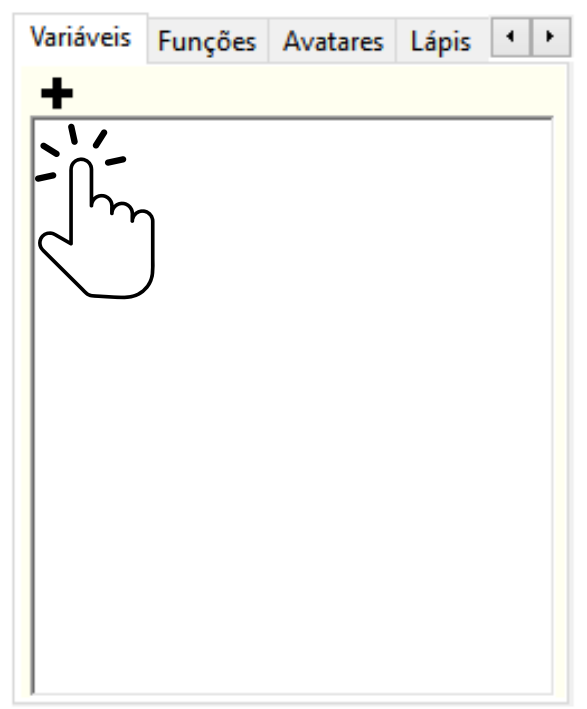


Mas, o que são variáveis?

Variáveis são espaços na memória reservados para armazenar informações que podem mudar durante a execução de um programa. Elas permitem que dados sejam guardados, manipulados e reutilizados ao longo do código.


Para criar variáveis na GenIA basta seguir as seguintes etapas:

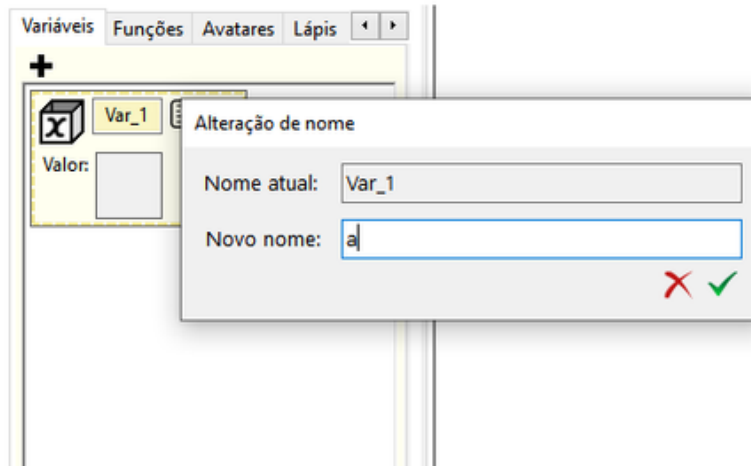
Na aba variáveis clicar no símbolo de "+".




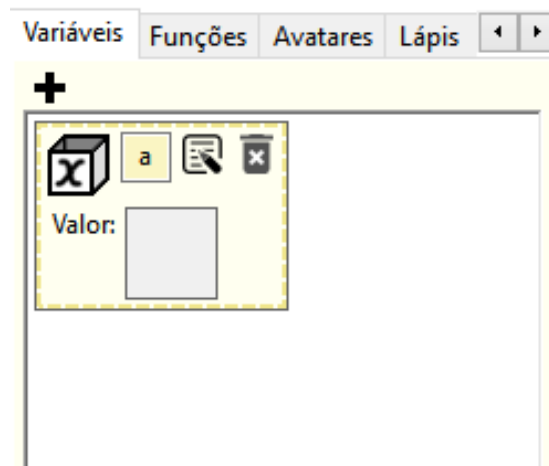
A variável será criada. Depois disso, podemos alterar seu nome para algo que facilite identificar o que ela armazena.

CRIANDO VARIÁVEIS

Para alterar o nome da variável basta clicar no botão 'Editar nome'  ao lado do nome da variável.



Digitar o novo nome para a variável e confirmar a alteração no botão verde .

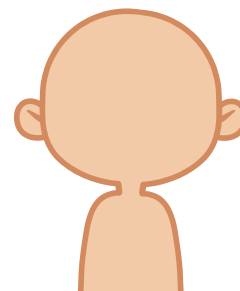


ADICIONANDO AVATARES



[Clique aqui para assistir o tutorial.](#)

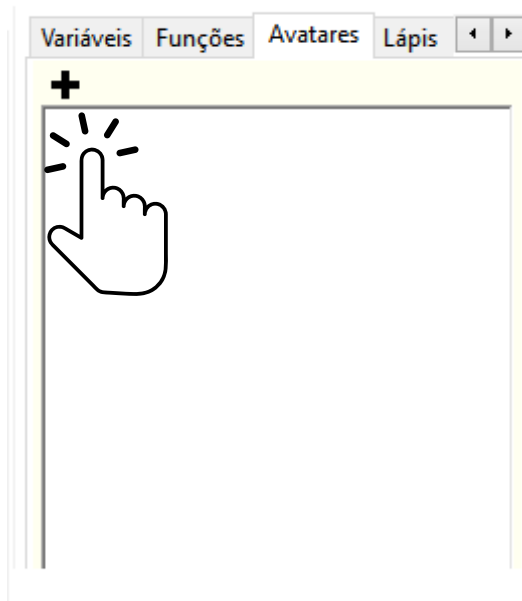
Os avatares são personagens que fazem a interação com o usuário durante a execução da programação.




A adição dos avatares pode ocorrer de duas formas, uma utilizando o avatar existente na GenIA (o gatinho Pi) e outra carregando imagens diretamente do computador.

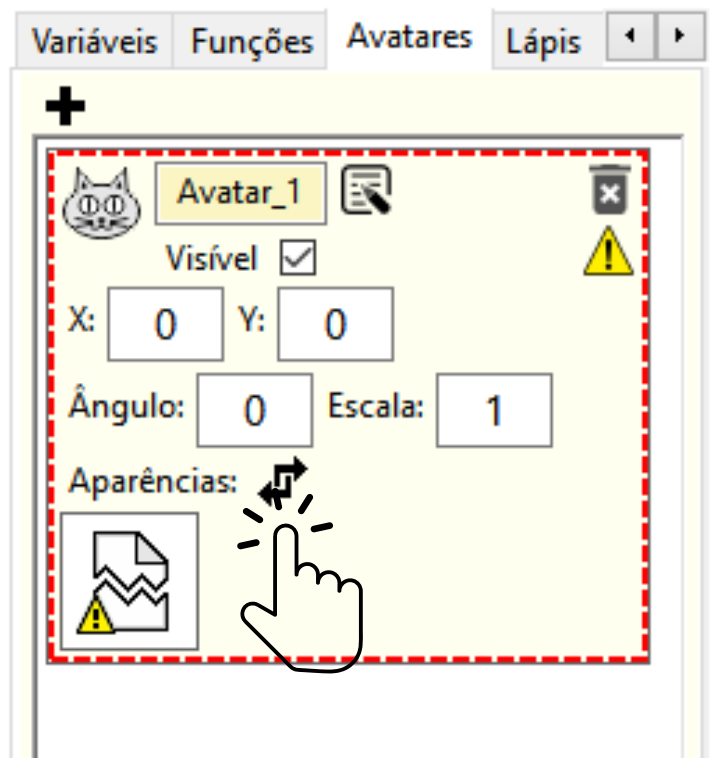
Vamos apresentar as duas formas para conhecimento.

Para adicionar os avatares, iniciaremos indo na aba Avatares e clicando no símbolo de '+'.
+
+



ADICIONANDO AVATARES

Após isso, inserir a imagem desejada clicando no símbolo .

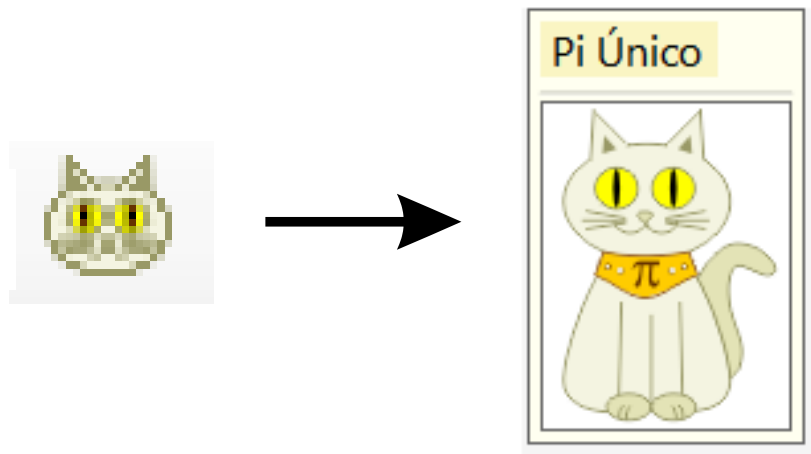


Para adicionar o Pi, basta clicar na imagem de gatinho conforme figura abaixo:




ADICIONANDO AVATARES

Existem duas formas de avatares do Pi: ele de frente, conforme podemos observar abaixo:



E ele simulando um movimento, conforme podemos observar abaixo:

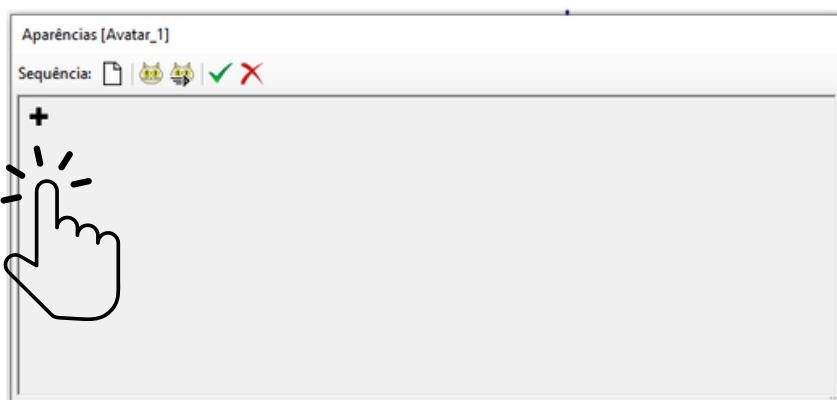



Basta, agora confirmar a escolha clicando no botão de confirmar alterações .

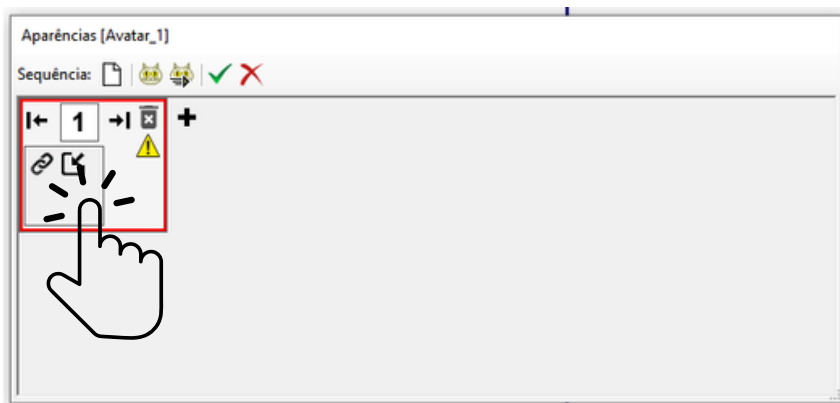
ADICIONANDO AVATARES

Para adicionar uma imagem diretamente do computador como avatar, é necessário que o arquivo esteja salvo localmente. Caso deseje, você pode realizar o download da imagem no próprio computador onde a GenIA está instalada, antes de adicioná-la à plataforma.

Em seguida, na caixa de inclusão de avatares, clique no botão '+' para iniciar o processo de adição.

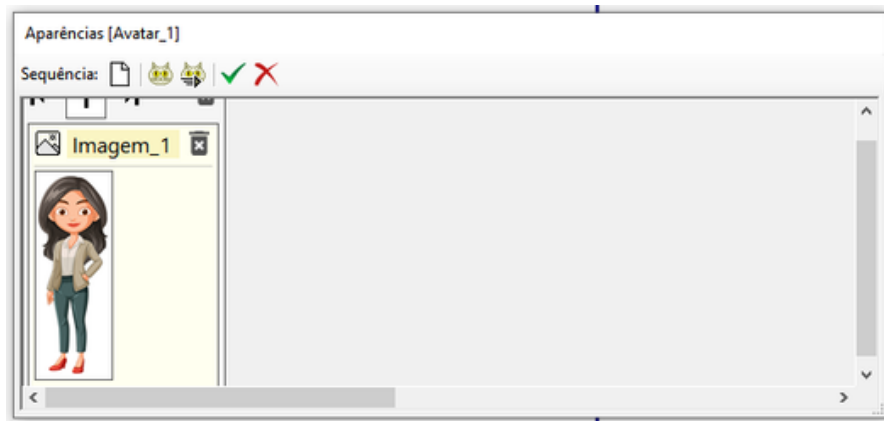



Após a abertura da caixa do novo avatar, basta clicar no botão importar uma imagem como uma nova aparência .



ADICIONANDO AVATARES

E então escolher a imagem desejada como novo avatar no drive do computador.



Concluindo o processo, basta então confirmar a alteração clicando no botão .

ESTRUTURAS CONDICIONAIS



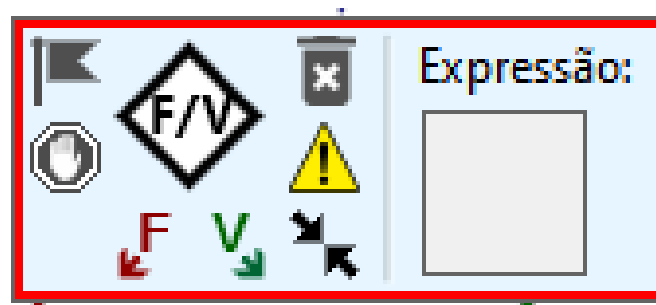
[Clique aqui para assistir o tutorial.](#)

Estrutura condicional é um recurso da programação que permite tomar decisões durante a execução de um programa. Ela funciona como uma escolha: dependendo de uma condição ser verdadeira ou falsa, o programa executa um determinado conjunto de comandos.

Para criar uma estrutura condicional na GenIA basta ir em comandos, aba Geral, escolher "Decidir" e arrastar até o fluxograma.



No comando "Decidir", é necessário inserir a expressão que representa a condição. A partir dela, o programa seguirá um caminho diferente dependendo se essa condição for falsa ou verdadeira.

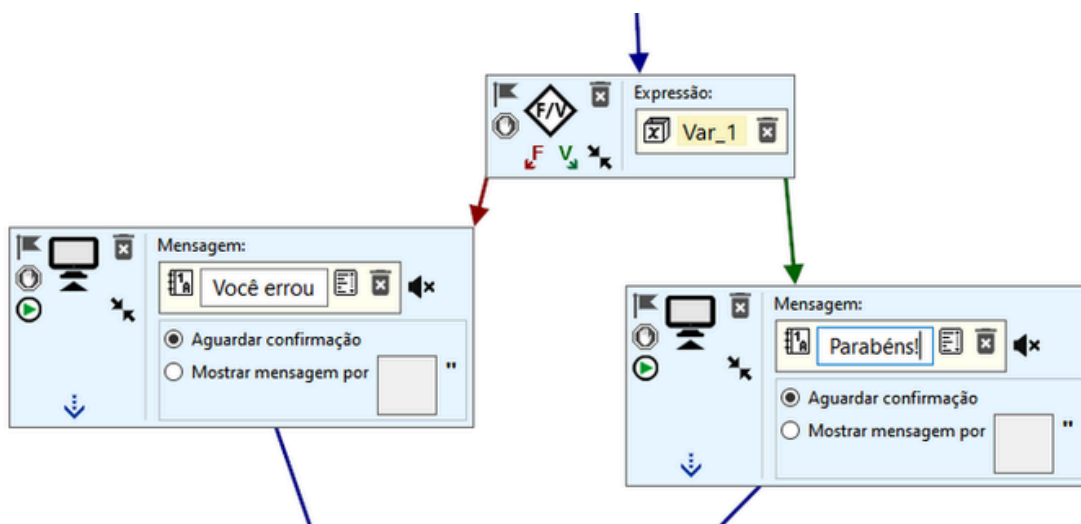


ESTRUTURAS CONDICIONAIS

A expressão pode ser formada simplesmente por uma variável, caso ela armazene uma informação do tipo lógica, ou seja, SIM/NÃO ou VERDADEIRO/FALSO. Nesse caso, o programa seguirá para a saída correspondente ao valor armazenado nessa variável.

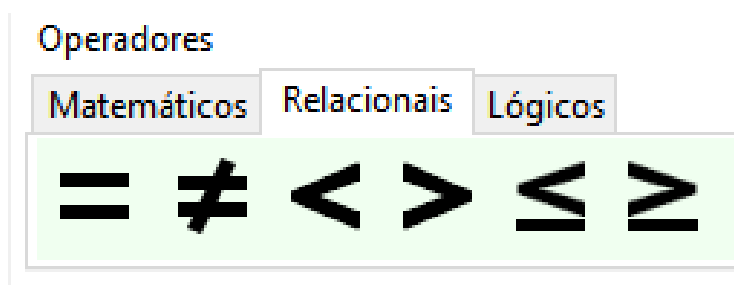
Alternativamente, a expressão pode ser construída utilizando operadores matemáticos ou relacionais, como $>$, $<$, $=$, entre outros, permitindo condições mais complexas.

Para usar com variável com informação do tipo lógica basta arrastar essa variável ao campo expressão e seguir com os comandos caso seja verdadeiro ou falso.

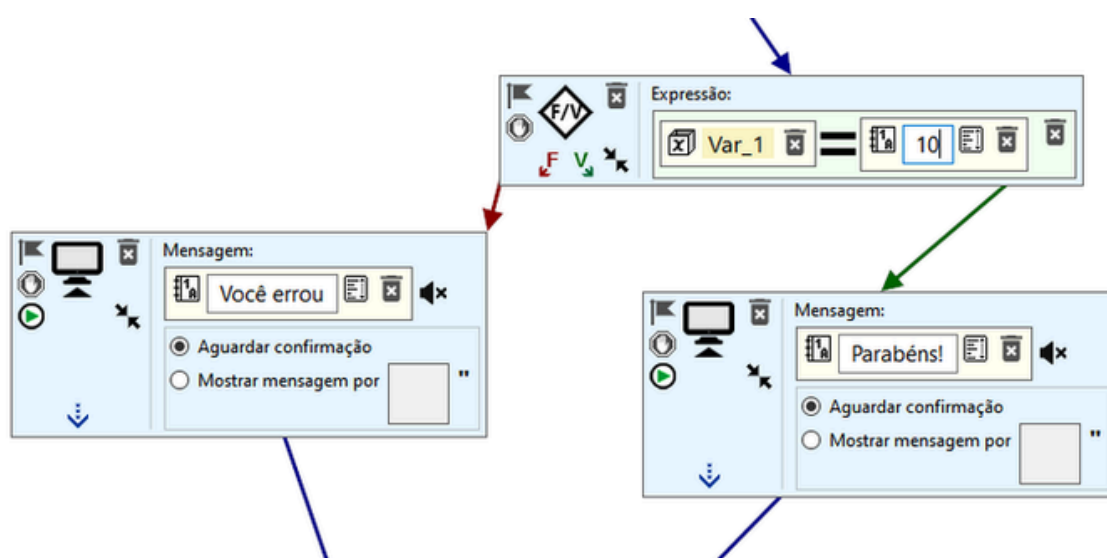


ESTRUTURAS CONDICIONAIS

Para usar com operadores relacionais, basta ir até os operadores, aba "Relacionais" e arrastar até a expressão o operador que deseja utilizar.



Após isso, basta inserir os dados da expressão, podendo adicionar variáveis e números para comparar e seguir com os comandos caso a expressão seja verdadeira ou falsa.



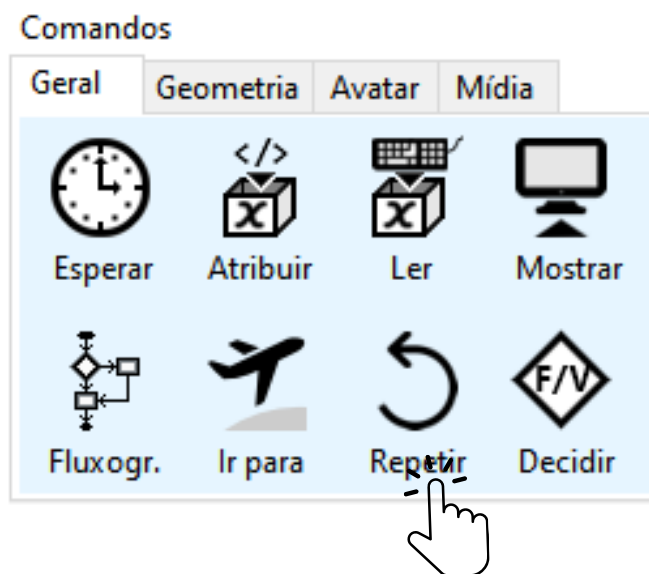
ESTRUTURAS DE REPETIÇÃO



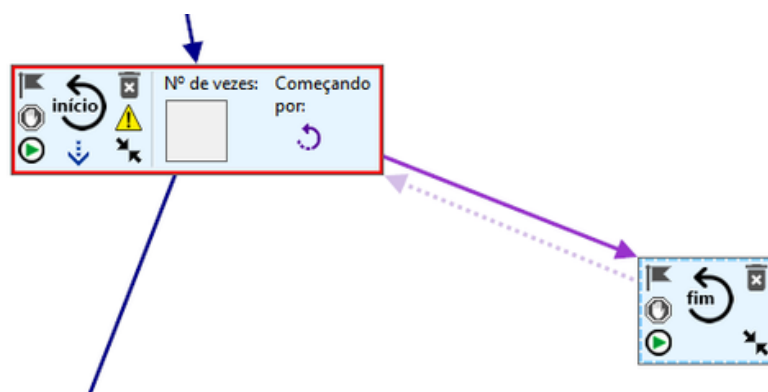
[Clique aqui para assistir o tutorial.](#)

Estruturas de repetição, também chamadas de laços ou loops, são comandos da programação que permitem repetir um conjunto de instruções várias vezes, de forma automática. Elas são muito importantes, pois evitam que o programador tenha que escrever o mesmo comando várias vezes.

Para criar uma estrutura de repetição na GenIA basta ir em comandos, aba Geral, escolher "Repetir" e arrastar até o fluxograma.

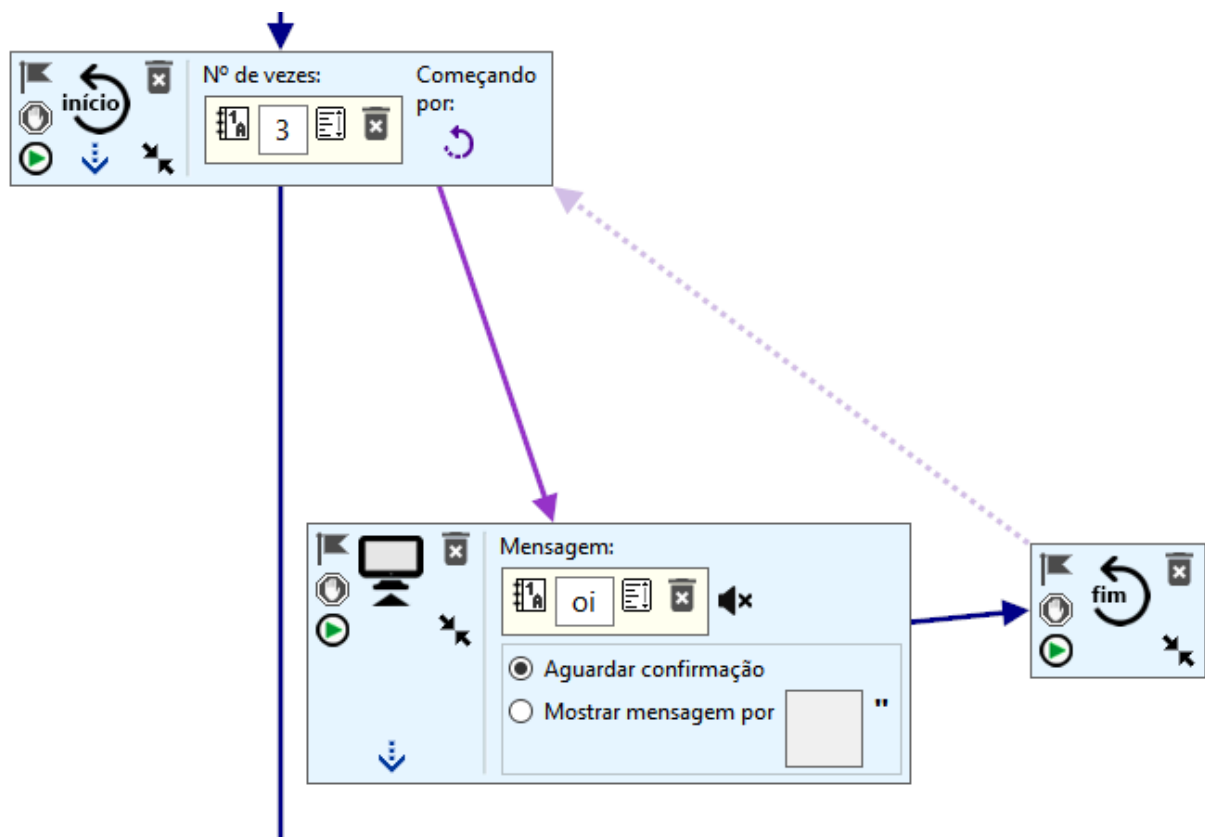


No comando "Repetir", é necessário inserir a um número de vezes que os passos serão repetidos.



ESTRUTURAS DE REPETIÇÃO

Os comandos que devem ser repetidos são aqueles inseridos antes da instrução "Fim". Após concluir o número de repetições definidas, o programa seguirá automaticamente para o próximo passo.



No exemplo acima, o texto "oi" será exibido três vezes consecutivas, e, em seguida, o programa continuará com a execução dos demais comandos.

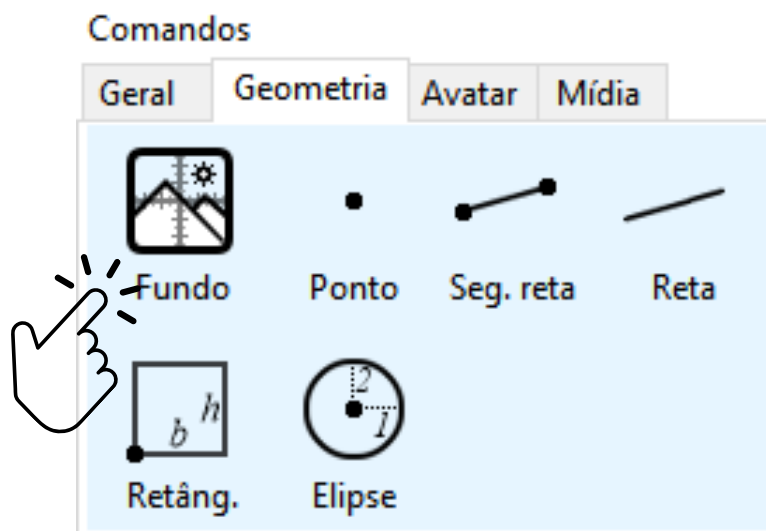


DICAS NA PROGRAMAÇÃO

Sempre inicie a programação definindo um fundo, pois ele estabelece o contexto e o objetivo do Objeto de Aprendizagem (OA). Esse fundo pode ser uma imagem, uma cor sólida ou, por exemplo, um plano cartesiano, caso o tema envolva conteúdos relacionados.

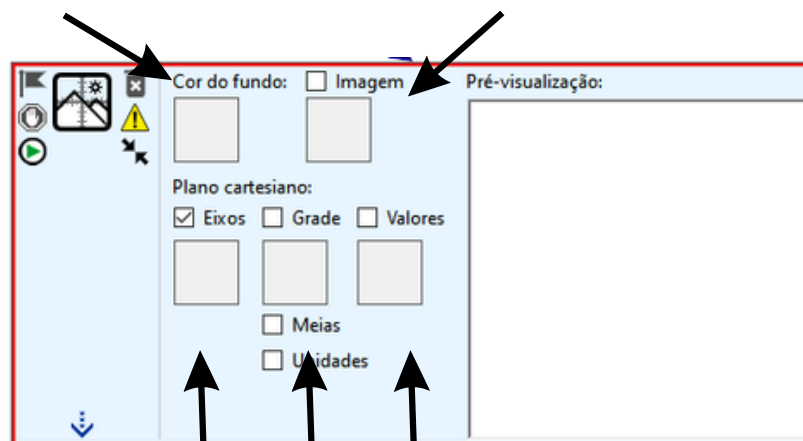
Para inserir o fundo, acesse a aba "Geometria" na seção de comandos e selecione a opção "Fundo".

Lembre-se: a imagem ou o tipo de fundo pode ser alterado a qualquer momento durante a programação.



Para colocar cor de fundo

Para adicionar imagem



Para deixar formato de plano cartesiano

03

PLANEJAMENTO DE AULAS PLUGADAS

Aulas plugadas são aquelas em que o uso do computador é integrado às atividades, explorando recursos digitais para apoiar o processo de aprendizagem.

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Conhecendo a plataforma GenIA

Foco e objetivos da aula:

- Compreender a programação da GenIA.
- Fazer programação de maneira orientada.

Materiais necessários:

- Computador com GenIA instalada.

Nexos conceituais de Navarro:

Pensamento algébrico: Exploração dos passos para caminhar até o ponto que se deseja.

Pensamento algorítmico: Compreender como montar o passo a passo para executar ação de caminhada e conversa.

Resolução de problemas: Como fazer um avatar interagir na tela.

Pilares de PC segundo Brackmann:

Abstração: Cada passo para a interação.

Decomposição: Como realizar e quais dados são necessários em cada passo da interação

Reconhecimento de padrões: Repetição da movimentação do Pi.

Algoritmo: Construção do fluxograma para a programação.

Estrutura / Atividade:

(Tempo: 50 minutos)

1. Acesse a plataforma GenIA
2. Pedir para que os estudantes adicionem o avatar pi.
3. Posicionar o Pi na coordenada (-10, 0)
4. Inserir o comando Andar da aba Avatar no fluxograma.
5. Preencher os dados do comando: incluindo o Pi como avatar e 10 passos.
6. Inserir o comando Interagir da aba Avatar no fluxograma.
7. Preencher os dados do comando: incluindo o Pi como avatar e informando uma fala de escolha dos estudantes.
8. Inserir o comando Virar da aba Avatar no fluxograma.
9. Preencher os dados do comando: incluindo o Pi como avatar e ângulo de 10 graus à direita.
10. Inserir o comando Andar da aba Avatar no fluxograma.
11. Preencher os dados do comando: incluindo o Pi como avatar e 30 passos.

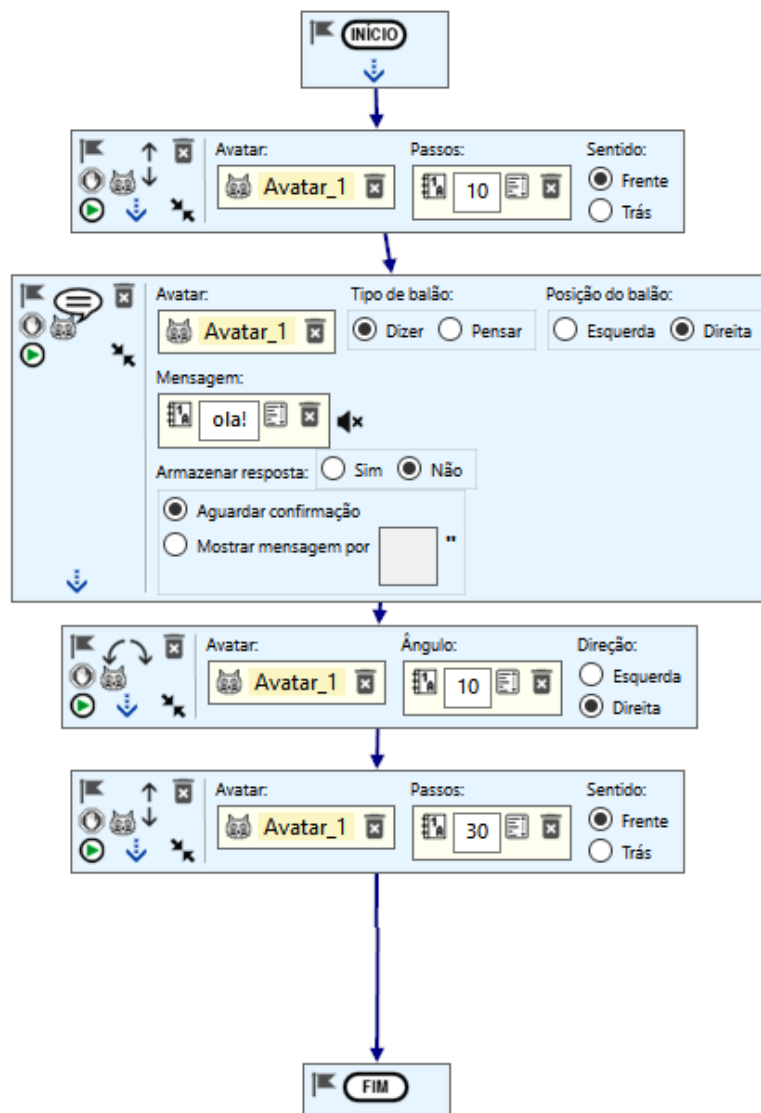
Avaliação:

A avaliação será realizada por meio da correção da folha de atividade preenchida pelos estudantes, com base no gabarito fornecido pelo professor. Serão considerados os seguintes critérios:

- Conversão correta das letras das frases para código binário (8 bits por caractere).
- Preenchimento completo da folha de atividade.
- Organização e legibilidade das respostas.
- Participação e empenho durante a realização da tarefa.

A devolutiva poderá ser feita em sala de aula, com correção coletiva e esclarecimento de eventuais dúvidas sobre a conversão binária.

FLUXOGRAMA A SER CONSTRUÍDO



PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Conhecendo código binário

Foco e objetivos da aula:

- Compreender como um computador se comunica
- O que são Bit e Byte
- Conceitos básicos de programação

Materiais necessários:

- Computador com GenIA instalada.
- Objeto de aprendizagem: Código binário (Letícia C Carvalho)_(zip).
- Folha de atividades

Nexos conceituais de Navarro:

Pensamento algébrico: Identificando os padrões binários que pertencem a cada letra do alfabeto.

Pensamento algorítmico: Criando estratégia para realizar a conversão das palavras em binário.

Resolução de problemas: Transcrever em código binário as frases disponíveis na folha.

Pilares de PC segundo Brackmann:

Abstração: Abstraindo cada letra da frase, ignorando seu sentido e a palavra.


Decomposição: Decompondo a frase em palavras e letras.


Reconhecimento de padrões: Reconhecendo regularidades no código binário.

Algoritmo: Passo a passo para conversão.

Estrutura / Atividade:

(Tempo: 50 minutos)

Após a conceituação de bit e byte em sala de aula (de acordo com a aula desplugada ) , os estudantes realizarão uma atividade prática de fixação.

1. Acesse o site www.plataformagenia.com  e, na aba Objetos de Aprendizagem, faça o download do Código Binário (Letícia C. Carvalho) em formato ZIP.
2. Com o objeto de aprendizagem aberto na plataforma GenIA, os estudantes deverão utilizar a ferramenta para converter em código binário as frases indicadas na folha de atividade fornecida pelo professor.
3. Os resultados das conversões deverão ser registrados na própria folha, preenchendo todos os campos conforme orientações.



Sempre que encontrar uma seta em uma frase, isso indica que o item é clicável e levará você diretamente ao conteúdo correspondente.

Avaliação:

A avaliação será realizada por meio da correção da folha de atividade preenchida pelos estudantes, com base no gabarito fornecido pelo professor. Serão considerados os seguintes critérios:

- Conversão correta das letras das frases para código binário (8 bits por caractere).
- Preenchimento completo da folha de atividade.
- Organização e legibilidade das respostas.
- Participação e empenho durante a realização da tarefa.

A devolutiva poderá ser feita em sala de aula, com correção coletiva e esclarecimento de eventuais dúvidas sobre a conversão binária.

FOLHA DE ATIVIDADES

Nome do estudante:

Série:

Disciplina: Pensamento Computacional

Converta em Código Binário as seguintes frases, utilizando a tabela ASCII (Ignorar os acentos e o espaço).

1) Livro é bom

2) O Céu é azul

3) Amo animais de estimação

4) Sou inteligente

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Exploração livre

Foco e objetivos da aula:

- Compreenderos recursos da GenIA

Materiais necessários:

- Computador com GenIA instalada.

Nexos conceituais de Navarro:

Pensamento algébrico: Compreender repetições e possíveis variáveis a serem utilizadas.

Pensamento algorítmico: Como estruturar o que desejam criar.

Resolução de problemas: Definição no planejamento dos estudantes.

Pilares de PC segundo Brackmann:

Abstração: Perceber quais informações do que se deseja fazer são importantes.

Decomposição: Pensar em cada passo para atingir o objetivo que se deseja.

Reconhecimento de padrões: O que se repete no que pretende fazer.

Algoritmo: Fluxograma programado.

Estrutura / Atividade:

(Tempo: 50 minutos)

1. Acesse a plataforma GenIA.
2. Os estudantes deverão planejar o que desejam programar.
3. Deixar os estudantes explorar a GenIA de forma livre.

Avaliação:

A avaliação será realizada por meio da observação em sala de aula e entrega da programação. Serão considerados os seguintes critérios:

- Lógica criada
- Funcionamento da programação.

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Compreendendo Plano Cartesiano

Foco e objetivos da aula:

- Compreender como identificar as coordenadas de um plano cartesiano.

Materiais necessários:

- Computador com GenIA instalada.

Nexos conceituais de Navarro:

Pensamento algébrico: Compreendendo as variáveis necessárias na programação.

Pensamento algorítmico: Criando o passo a passo na programação para gerar um ponto no plano cartesiano.

Resolução de problemas: Criar um Objeto de aprendizagem capaz de identificar o ponto em um plano cartesiano na GenIA.

Pilares de PC segundo Brackmann:

Abstração: Identificando os dados necessários da coordenada.


Decomposição: Localizar o ponto primeiro na horizontal e depois na vertical.

Reconhecimento de padrões: Reconhecendo que o eixo X é na horizontal e Y na vertical.

Algoritmo: Passo a passo para criação da programação.

Estrutura / Atividade:

(Tempo: 50 minutos)

Após a conceituação de plano cartesiano em sala de aula (de acordo com a aula desplugada ) os estudantes realizarão uma atividade prática de fixação.

1. Acesse a plataforma GenIA.
2. Solicite que os estudantes criem uma programação para determinar um ponto no plano cartesiano com base nas coordenadas fornecidas pelo usuário.
3. Primeiramente, os estudantes devem identificar as variáveis necessárias para a programação.
4. Em seguida, deverão compreender a sequência das coordenadas no plano cartesiano: primeiro, obter o valor de X e, depois, o valor de Y.



Sempre que encontrar uma seta em uma frase, isso indica que o item é clicável e levará você diretamente ao conteúdo correspondente.

Avaliação:

A avaliação será realizada por meio da observação em sala de aula e entrega da programação. Serão considerados os seguintes critérios:

- Correta sequência da atividade (algoritmo correto).
- Funcionamento da programação.

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Compreendendo Algoritmo

Foco e objetivos da aula:

- Compreender o que é algoritmo através de uma sequência de conversa.

Materiais necessários:

- Computador com GenIA instalada.

Nexos conceituais de Navarro:

Pensamento algébrico: Percebendo que os diálogos seguem uma estrutura (saudação, pergunta, resposta, despedida).

Pensamento algorítmico: Pensando em como será criada a sequência do diálogo.

Resolução de problemas: Criar a programação de uma sequência de diálogo.

Pilares de PC segundo Brackmann:

Abstração: Ordem do diálogo, condições que fazem a conversa avançar.


Decomposição: Falas de cada um.

Reconhecimento de padrões: Identificar conversas previsíveis como saudações.

Algoritmo: O passo a passo do diálogo.

Estrutura / Atividade:

(Tempo: 50 minutos)

Após a conceituação de algoritmo em sala de aula (de acordo com a atividade desplugada ) , os estudantes realizarão uma atividade prática de fixação.

1. Acesse a plataforma GenIA.
2. Solicite que os estudantes criem uma programação com diálogo entre dois ou mais avatares na GenIA



Sempre que encontrar uma seta em uma frase, isso indica que o item é clicável e levará você diretamente ao conteúdo correspondente.

Avaliação:

A avaliação será realizada por meio da observação em sala de aula e entrega da programação. Serão considerados os seguintes critérios:

- Correta sequência da atividade (algoritmo correto).
- Funcionamento da programação.

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Compreendendo Estruturas Condicionais

Foco e objetivos da aula:

- Compreender como funciona uma estrutura condicional

Materiais necessários:

- Computador com GenIA instalada.

Nexos conceituais de Navarro:

Pensamento algébrico: Na criação da expressão para determinar a condição.

Pensamento algorítmico: Sabendo organizar o passo a passo da programação.

Resolução de problemas: Identificar tomada de decisão.

Pilares de PC segundo Brackmann:

Abstração: Identificar o que importa para tomada de decisão.


Decomposição: Identificar o que será Verdadeiro ou Falso.

Reconhecimento de padrões: Compreender quando é necessário a decisão.

Algoritmo: O passo a passo envolvendo condicional.

Estrutura / Atividade:

(Tempo: 50 minutos)

Após a conceituação de estruturas condicionais em sala de aula (de acordo com a aula desplugada ) , os estudantes realizarão uma atividade prática de fixação.

1. Acesse a plataforma GenIA.
2. Solicite que os estudantes criem uma programação envolvendo operações matemáticas. Caso a conta esteja errada, ele precisa avisar que está errada, caso esteja correta, deverá parabenizar o usuário.



Sempre que encontrar uma seta em uma frase, isso indica que o item é clicável e levará você diretamente ao conteúdo correspondente.

Avaliação:

A avaliação será realizada por meio da observação em sala de aula e entrega da programação. Serão considerados os seguintes critérios:

- Correta sequência da atividade (algoritmo correto).
- Funcionamento da programação.

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Compreendendo Estruturas de Repetição

Foco e objetivos da aula:

- Compreender como funciona uma estrutura condicional

Materiais necessários:

- Computador com GenIA instalada.

Nexos conceituais de Navarro:

Pensamento algébrico: Identificar os padrões para criar o looping.

Pensamento algorítmico: Sabendo organizar o passo a passo da programação.

Resolução de problemas: Como podemos criar um cronometro na GenIA?

Pilares de PC segundo Brackmann:

Abstração: Identificar onde o looping será criado.


Decomposição: Identificar os comandos que ficam fora e dentro da estrutura de repetição.

Reconhecimento de padrões: Compreender o padrão que se repete.

Algoritmo: O passo a passo.

Estrutura / Atividade:

(Tempo: 50 minutos)

Após a conceituação de estruturas de repetição em sala de aula (de acordo com a aula desplugada ) , os estudantes realizarão uma atividade prática de fixação.

1. Acesse a plataforma GenIA.
2. Solicite que os estudantes criem uma programação de um cronômetro que conte até 60 segundos. Na próxima página tem o gabarito do algoritmo.

Um exemplo de algoritmo de cronometro está disponível no site www.plataformagenia.com na aba objetos de aprendizagem.

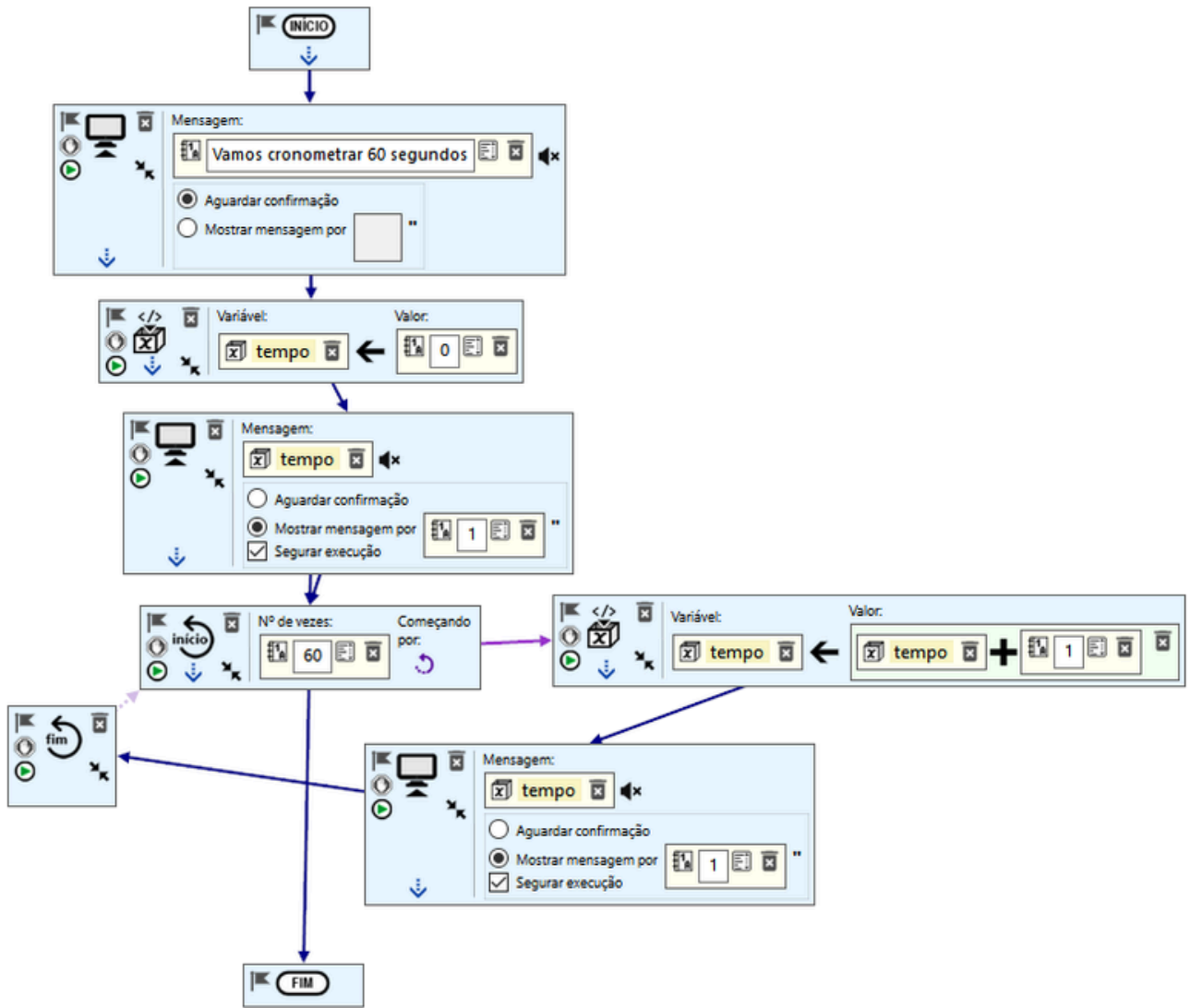


Sempre que encontrar uma seta em uma frase, isso indica que o item é clicável e levará você diretamente ao conteúdo correspondente.

Avaliação:

A avaliação será realizada por meio da observação em sala de aula e entrega da programação. Serão considerados os seguintes critérios:

- Correta sequência da atividade (algoritmo correto).
- Funcionamento da programação.



04

PLANEJAMENTO DE AULAS DESPLUGADAS

Aulas desplugadas são realizadas em sala de aula sem o uso de computadores, utilizando recursos físicos e dinâmicas manuais para ensinar conceitos de programação e pensamento computacional.

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Entendendo algoritmo em descrição narrativa

Foco e objetivos da aula:

- Compreender o que é algoritmo

Materiais necessários:

- Quadro
- Giz

Nexos conceituais de Navarro:

Pensamento algébrico: Etapas lógicas e trabalho com quantidades de ingredientes.

Pensamento algorítmico: Pensar em cada etapa da receita.

Resolução de problemas: Como podemos fazer um bolo?

Pilares de PC segundo Brackmann:

Abstração: Analisar o que é importante na receita.

Decomposição: Ação sobre cada ingrediente.

Reconhecimento de padrões: Reconhecer quais ações se repetem, como: pegar, assar, mexer, e entre outros.

Algoritmo: A receita em si.

Estrutura / Atividade:

(Tempo: 20 minutos)

1. Passar no quadro o conceito de algoritmo:

Um algoritmo é uma sequência organizada de passos utilizada para realizar uma tarefa ou resolver um problema.

Ele pode ser representado de três formas principais:

- Descrição narrativa: o passo a passo é escrito em linguagem natural, com detalhes claros sobre cada etapa;
- Pseudocódigo: utiliza uma linguagem estruturada próxima à programação, mas ainda legível para humanos, facilitando a transição para o código real;
- Representação gráfica: apresenta o algoritmo visualmente, por meio de formas e setas, como em um fluxograma, facilitando a compreensão do fluxo de execução.

(Tempo: 30 minutos)

2. Realizar com os estudantes, no quadro, a construção coletiva de um passo a passo para fazer um bolo. Reforce que receitas e manuais de instruções são exemplos práticos de algoritmos. Durante a atividade, peça que os alunos indiquem as etapas necessárias, enquanto a professora registra no quadro, respeitando a ordem lógica da sequência

Avaliação:

A avaliação será realizada por meio da observação em sala de aula. Serão considerados os seguintes critérios:

- Correta sequência da atividade (algoritmo correto).

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Conceituando fluxograma

Foco e objetivos da aula:

- Compreender o uso de fluxograma
- Compreender como criar fluxograma

Materiais necessários:

- Folha com exemplo de fluxograma e símbolos
- Folha em Branco
- Folha com descrição narrativa

Nexos conceituais de Navarro:

Pensamento algébrico: Transição do pensamento concreto para o simbólico na transcrição da atividade.

Pensamento algorítmico: Diferença nas sequencias do algoritmo em descrição narrativa e fluxograma.

Resolução de problemas: Como podemos transcrever uma descrição em fluxograma?

Pilares de PC segundo Brackmann:

Abstração: selecionar apenas as informações relevantes da descrição narrativa

Decomposição: Dividir o texto narrativo em etapas menores (ações ou decisões).

Reconhecimento de padrões: Símbolos que se repetem.

Algoritmo: O fluxograma desenvolvido.

Estrutura / Atividade:

(Tempo: 20 minutos)

- Distribuir a folha com o exemplo de fluxograma e os principais símbolos utilizados. Explicar aos estudantes o funcionamento do fluxograma e como identificar os símbolos dentro do fluxo de execução.

(Tempo: 30 minutos)

- Entregar uma segunda atividade: uma folha em branco e outra folha contendo um algoritmo descrito em linguagem narrativa. Os estudantes deverão ler cada etapa da descrição narrativa e transcrevê-la para a forma gráfica, construindo o fluxograma na folha em branco.

Essa atividade pode ser realizada em grupos.

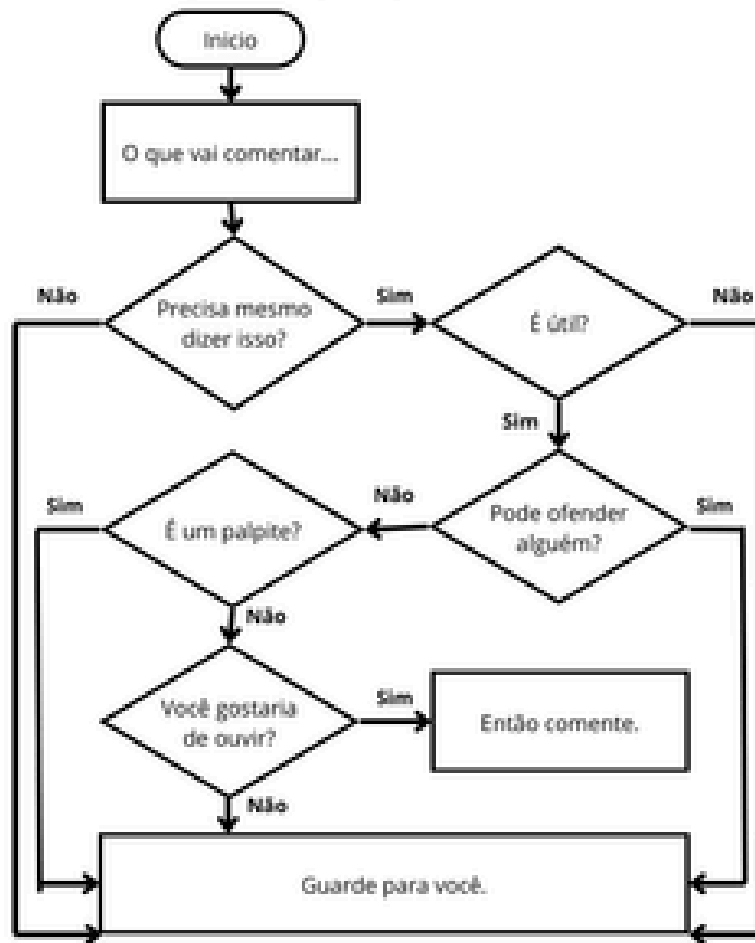
Avaliação:


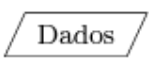



A avaliação será realizada por meio da entrega da atividade realizada pelos estudantes. Serão considerados os seguintes critérios:

- Correta sequência da atividade (algoritmo correto).
- Fluxograma com símbolos corretos.

EXEMPLO DE FLUXOGRAMA E SÍMBOLOS

Lógica simples para comentários



Símbolo	Função
	Indica o início e o fim do fluxograma
	Indica a entrada (input) e a saída (output) de dados
	Indica a execução/processamento de uma operação
	Uma pergunta é realizada para que a resposta (“Sim” ou “Não”) determine a sequência do fluxo
	Setas são usadas para conectar os passos do fluxograma indicando o fluxo da solução do problema

FOLHA DE ATIVIDADES

Nome do estudante:

Série:

Disciplina: Pensamento Computacional

Crie um fluxograma para o seguinte algoritmo de descrição narrativa:

1. Cachorro dá entrada no pet shop;
2. Recepcionista do pet shop verifica quais serviços a serem executados; Se tosa e banho, segue o processo 3 em diante, caso apenas banho, segue para o processo 4;
3. Cachorro é levado para a tosa
4. Cachorro é levado para o banho;
5. Após banho, cachorro é encaminhado para secagem;
6. Cachorro é encaminhado para antipulga e colocação de coleira e adereços;
7. Cachorro é encaminhado ao canil;
8. Devolução ao dono.

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Bit e Byte - compreendendo código binário

Foco e objetivos da aula:

- Compreender como o computador compreende as linguagens.
- Compreender o que são Bit e Byte.
- Compreender código binário

Materiais necessários:

- Quadro
- Giz

Nexos conceituais de Navarro:

Pensamento algébrico: Decodificando cada conjunto de 8 bits em caracter.

Pensamento algorítmico: Realizando o passo a passo para a decodificação.

Resolução de problemas: Qual é o frase por trás dos códigos binários?

Pilares de PC segundo Brackmann:

Abstração: Símbolo por trás da sequência.

Decomposição: Juntar de 8 em 8 conjunto de códigos binários.

Reconhecimento de padrões: Letra do alfabeto que equivale ao conjunto de 8 bits da frase.

Algoritmo: Passo a passo para a decodificação.

Estrutura / Atividade:

(Tempo: 30 minutos)

1) Passar no quadro o conceito de Bit, Byte e Código Binário:

"Bit" e "Byte" são termos relacionados à computação e representam diferentes conceitos de tamanho e armazenamento de dados.

BIT: A palavra "bit" é, na verdade, uma abreviação de "binary digit" (dígito binário) e tem suas raízes na eletrônica digital e na representação de dados em formato binário (0 ou 1). Cada "bit" corresponde a um sinal elétrico que pode estar em um dos dois estados, geralmente associados a "ligado" (1) ou "desligado" (0). Um "bit" é a menor unidade de informação em computação.

CODIGO BINÁRIO: O sistema binário ou de base 2 é um sistema de numeração posicional em que todas as quantidades se representam com base em dois números, ou seja, zero e um.

BYTE: Um "byte" é uma unidade de armazenamento de dados que consiste em um conjunto de 8 bits. Um byte pode representar um único caractere alfanumérico (como uma letra ou número) em uma codificação de caracteres, como o ASCII (American Standard Code for Information Interchange).

(Tempo: 20 minutos)

2) Passar o alfabeto da Tabela ASCII e uma frase em código Binário para a turma decodificar.

Avaliação:

A avaliação será realizada por meio da observação em sala de aula. Serão considerados os seguintes critérios:

- conclusão da atividade para decifrar a frase em binário proposta.

MATERIAL AUXILIAR

Tabela ASCII - O ASCII é um código que foi proposto por Robert W. Bemer como uma solução para unificar a representação de caracteres alfanuméricos em computadores. Antes de 1960, cada computador utilizava uma regra diferente para representar estes caracteres e o código ASCII nasceu para se tornar comum entre todas as máquinas.

A tabela do alfabeto, por exemplo é composta da seguinte maneira:

A	01000001	N	01001110	a	01100001	n	01101110
B	01000010	O	01001111	b	01100010	o	01101111
C	01000011	P	01010000	c	01100011	p	01110000
D	01000100	Q	01010001	d	01100100	q	01110001
E	01000101	R	01010010	e	01100101	r	01110010
F	01000110	S	01010011	f	01100110	s	01110011
G	01000111	T	01010100	g	01100111	t	01110100
H	01001000	U	01010101	h	01101000	u	01110101
I	01001001	V	01010110	i	01101001	v	01110110
J	01001010	W	01010111	j	01101010	w	01110111
K	01001011	X	01011000	k	01101011	x	01111000
L	01001100	Y	01011001	l	01101100	y	01111001
M	01001101	Z	01011010	m	01101101	z	01111010

Perceba que há códigos diferentes para letras maiúsculas e minúsculas. Para esta atividade, utilize apenas a tabela com letras maiúsculas.

CODIGO BINÁRIO PARA ATIVIDADE:

01010110 01001111 01000011 01000101 01010011 01010011 01000001 01001111
01001101 01000001 01010010 01000001 01010110 01001001 01001100 01001000
01001111 01010011 01001111 01010011

tradução: VOCESSAOMARAVILHOSOS - VOCES SAO MARAVILHOSOS

Perceba que não usamos o byte de espaço e caracteres com acentos.

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Compreendendo coordenadas

Foco e objetivos da aula:

- Compreender como funciona o algoritmo para determinar o ponto em um plano cartesiano

Materiais necessários:

- Quadro
- Giz
- Folha em branco
- Folha de atividade

Nexos conceituais de Navarro:

Pensamento algébrico: Compreensão das variáveis utilizadas nas coordenadas X e Y.

Pensamento algorítmico: Aprender a ordem para resolver o problema proposto.

Resolução de problemas: Como criar um algoritmo para determinar o ponto em um plano cartesiano?

Pilares de PC segundo Brackmann:

Abstração: Olhar as informações relevantes de cada comando para a sequência.

Decomposição: Compreensão de cada etapa do algoritmo.

Reconhecimento de padrões: Repetições de comandos alterando eixo do plano.

Algoritmo: O passo a passo para determinação do ponto no plano cartesiano.

Estrutura / Atividade:

(Tempo: 20 minutos)

1) Apresente aos estudantes o conceito de plano cartesiano, explicando que o eixo horizontal é chamado de eixo das abscissas (eixo X), e o eixo vertical, de eixo das ordenadas (eixo Y). Em seguida, demonstre como localizar um ponto no plano, indicando que primeiro se observa a posição no eixo X e, depois, no eixo Y.

(Tempo: 30 minutos)

2) Entregue aos estudantes a folha com a atividade e uma folha em branco. Oriente-os a recortar os trechos do algoritmo representado em fluxograma (em ordem incorreta) e colá-los na folha em branco, reorganizando-os na sequência correta para determinar um ponto no plano cartesiano.

Avaliação:

A avaliação será realizada por meio da entrega da atividade realizada pelos estudantes. Serão considerados os seguintes critérios:

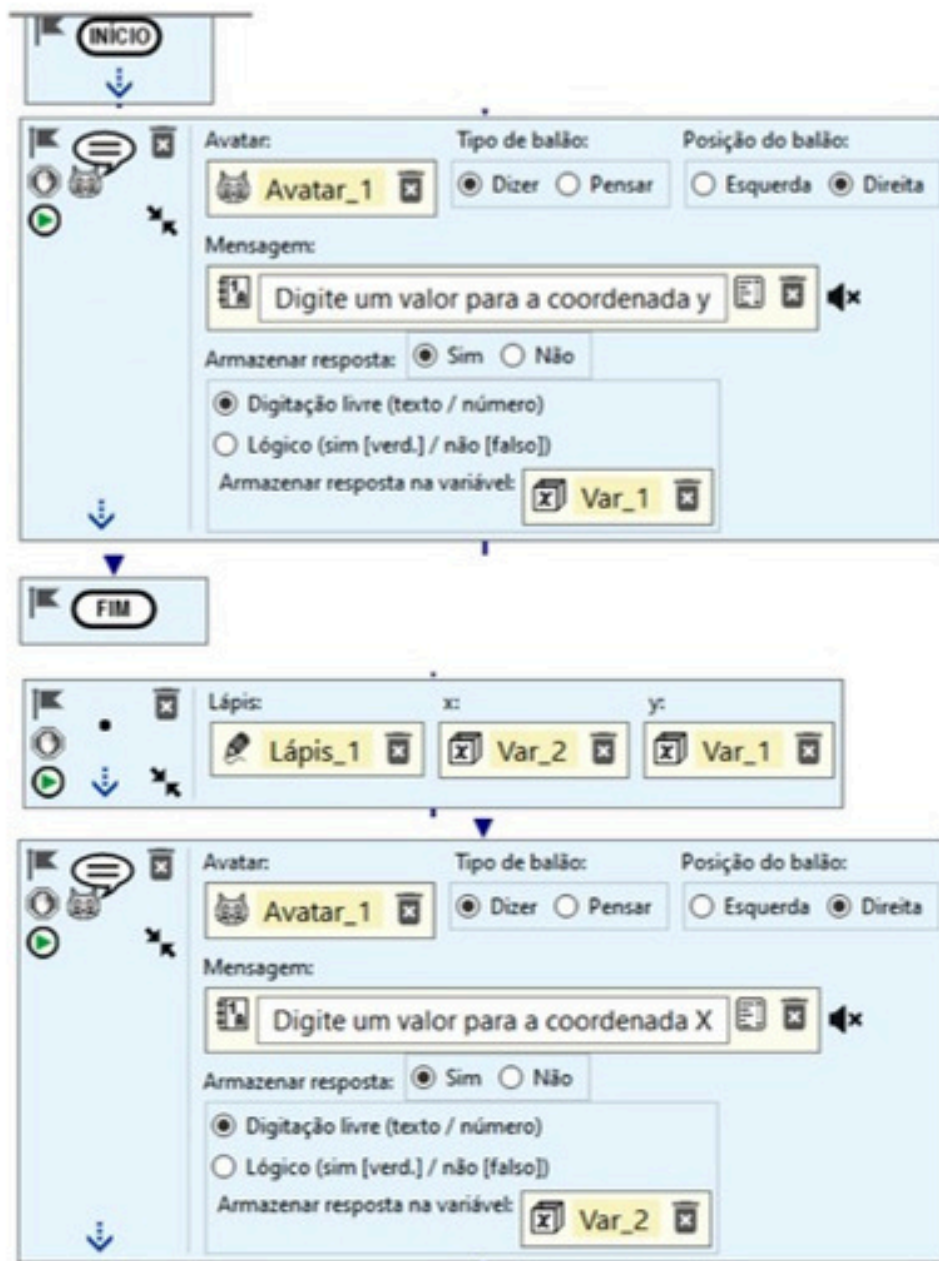
- Correta sequência da atividade (algoritmo correto).

FOLHA DE ATIVIDADES

Nome do estudante:

Série:

Disciplina: Pensamento Computacional



PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Compreendendo algoritmos

Foco e objetivos da aula:

- Compreender passo a passo através de um diálogo

Materiais necessários:

- Folha de atividade

Nexos conceituais de Navarro:

Pensamento algébrico: Percebendo que os diálogos seguem uma estrutura (saudação, pergunta, resposta, despedida).

Pensamento algorítmico: Pensando em como será criada a sequência do diálogo.

Resolução de problemas: Criar a programação de uma sequência de diálogo.

Pilares de PC segundo Brackmann:

Abstração: Ordem do diálogo, condições que fazem a conversa avançar.

Decomposição: Falas de cada um.

Reconhecimento de padrões: Identificar conversas previsíveis como saudações.

Algoritmo: O passo a passo do diálogo.

Estrutura / Atividade:

(Tempo: 20 minutos)

1) Explicar aos estudantes que um diálogo segue um algoritmo, pois é composto por uma sequência de passos a serem executados.

(Tempo: 30 minutos)

2) Distribuir a folha de atividade para os estudantes. Nela, eles deverão cortar as imagens dos rostos da Gretchen e do Neymar, que estão no início da folha, e colá-las nos locais dos avatares nos comandos de interação. Em seguida, os estudantes devem preencher os comandos com as falas de cada personagem, criando assim seu diálogo.

Avaliação:

A avaliação será realizada por meio da entrega da atividade realizada pelos estudantes. Serão considerados os seguintes critérios:

- Correta sequência da atividade (algoritmo correto).

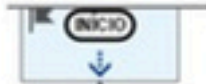
FOLHA DE ATIVIDADES

1/2

Nome do estudante:

Série:

Disciplina: Pensamento Computacional



Avatar: Tipo de balão: Dizer Pensar Posição do balão: Esquerda Direita

Mensagem:

Armazenar resposta: Sim Não

Aguardar confirmação

Mostrar mensagem por

Avatar: Tipo de balão: Dizer Pensar Posição do balão: Esquerda Direita

Mensagem:

Armazenar resposta: Sim Não

Aguardar confirmação

Mostrar mensagem por

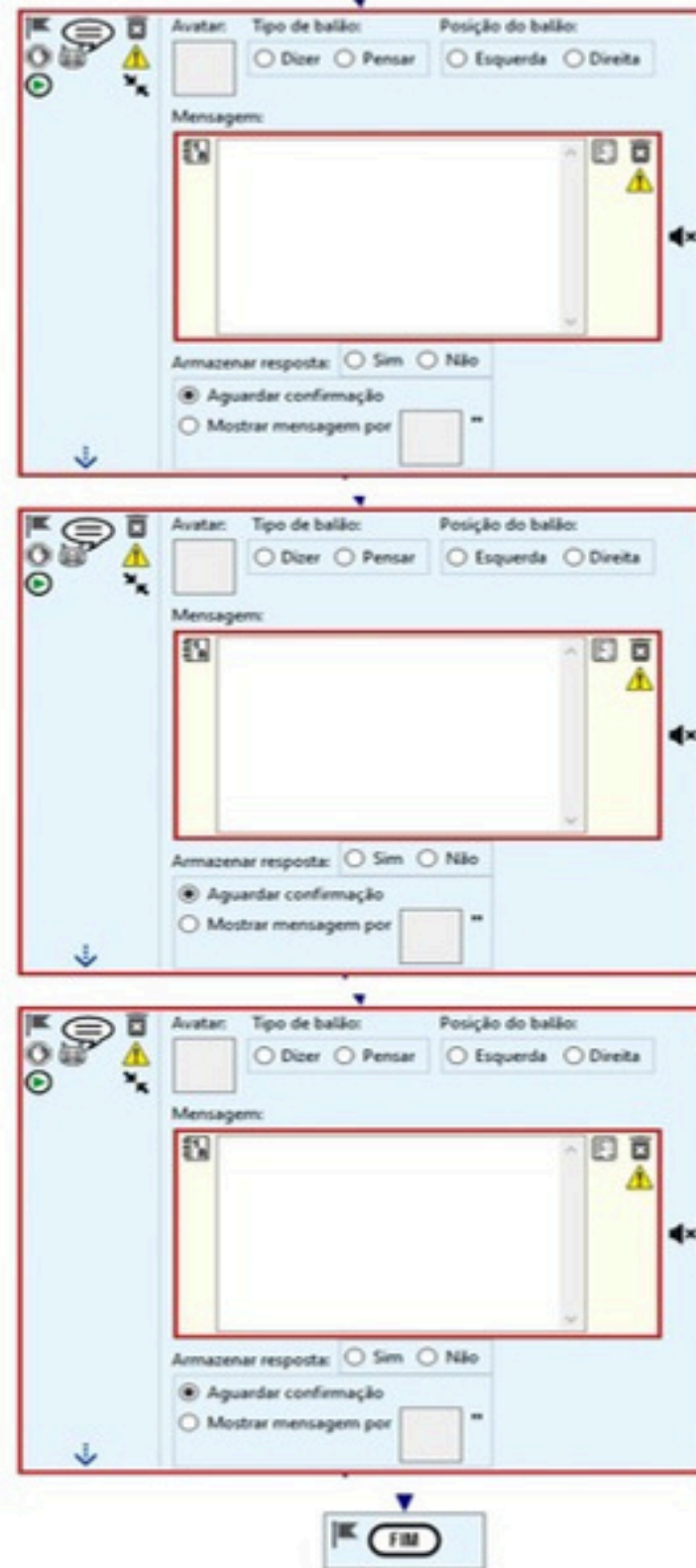
Avatar: Tipo de balão: Dizer Pensar Posição do balão: Esquerda Direita

Mensagem:

Armazenar resposta: Sim Não

Aguardar confirmação

Mostrar mensagem por



Realizar a atividade cortando os avatares do Neymar e Gretchen e colar nos respectivos locais montando uma conversa entre eles como se tivesse programando na GenIA

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Compreendendo estruturas condicionais

Foco e objetivos da aula:

- Compreender como funciona uma estrutura condicional

Materiais necessários:

- Folha de atividade

Nexos conceituais de Navarro:

Pensamento algébrico: Na estratégia para a tomada de decisão.

Pensamento algorítmico: Seguir a sequência do jogo.

Resolução de problemas: Encontrar estratégias para avançar.

Pilares de PC segundo Brackmann:

Abstração: Identificar o que importa para tomada de decisão.

Decomposição: Identificar a melhor estratégia.

Reconhecimento de padrões: Compreender quando é necessário a decisão e desafio.

Algoritmo: Os passos do jogo.

Estrutura / Atividade:

(Tempo: 10 minutos)

1) Explicar o que é uma estrutura condicional de acordo com o subtítulo:

estruturas condicionais  informado neste ebook.

(Tempo: 05 minutos)

2) Questionar os estudantes quando eles tomam decisões e como eles agem de acordo com ela no dia a dia.

(Tempo: 35 minutos)

2) Distribuir as folhas de atividade para os estudantes. Eles deverão recortar os quadrados com números e os cards.

- O jogo é um jogo de trilha, ganha o primeiro que chegar até o final.

- Os números recortados devem ficar embaralhados de cabeça para baixo para que o estudante retire o número de passos que deverá dar. Os bonecos do jogo podem ser uma bolinha de papel.

- Se o estudante chegar em uma das casas, desafio ou decisão deverá pegar uma carta respectiva a casa de cair. Na carta de decisão ele deverá escolher o caminho para seguir, na carta de desafio, ele deverá realizar o desafio antes de prosseguir com o jogo.



Sempre que encontrar uma seta em uma frase, isso indica que o item é clicável e levará você diretamente ao conteúdo correspondente.

Avaliação:

A avaliação será realizada por meio da entrega da atividade realizada pelos estudantes. Serão considerados os seguintes critérios:

- Avaliar se a estrutura condicional fez sentido para o estudante observando o trabalho em sala de aula.

1

2

3

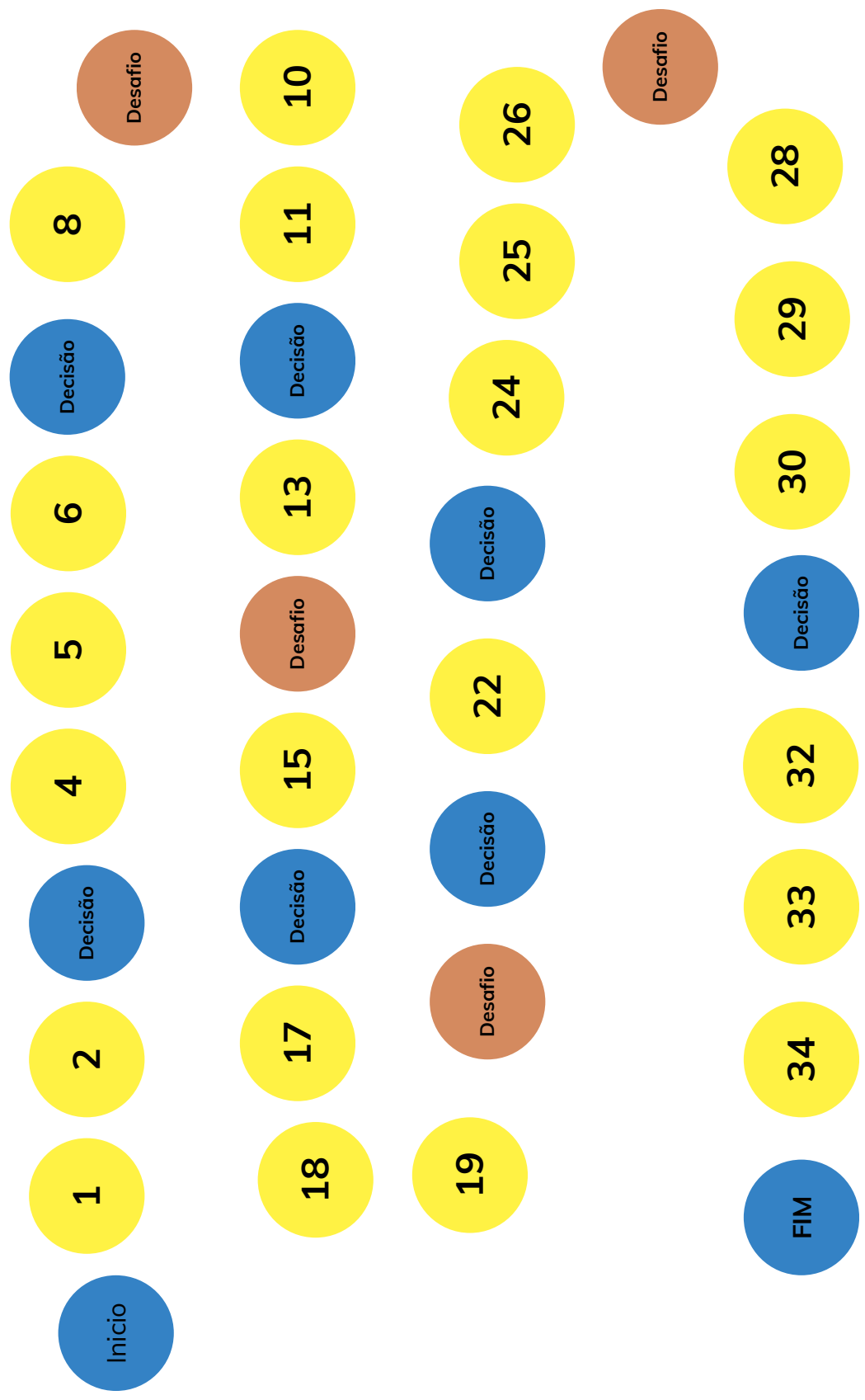
4

5

6

7

8



<p>Desafio</p> <p>Decomposição</p> <p>Quais são as menores etapas para “fazer um bolo”?</p>	<p>Desafio</p> <p>Decomposição</p> <p>Quais são as partes de um computador?</p>	<p>Desafio</p> <p>Algoritmo</p> <p>Descreva o passo a passo para escovar os dentes.</p>
<p>Desafio</p> <p>Reconhecimento de padrões</p> <p>Qual é o próximo número da sequência?</p> <p>3, 7, 11, ?</p>	<p>Desafio</p> <p>Abstração</p> <p>Para atravessar a rua, quais detalhes são essenciais?</p>	<p>Desafio</p> <p>Algoritmo</p> <p>Descreva o passo a passo para atravessar a rua.</p>
<p>Desafio</p> <p>Reconhecimento de padrões</p> <p>Qual é o próximo número da sequência?</p> <p>2, 10, 12, ?</p>	<p>Desafio</p> <p>Abstração</p> <p>Quais detalhes são importantes no problema:</p> <p>Tenho 2 lápis e uma borracha, quero comprar mais uma caneta, quantos materiais terei?</p>	<p>Decisão</p> <p>Você encontrou um atalho. sorteie mais um número para andar casas. Se a escolha for o número: par fique no lugar, se for ímpar ande mais duas casas.</p>
<p>Decisão</p> <p>Se você esta na frente do seu oponente, anda 1 casa, se não estiver fique no lugar.</p>	<p>Decisão</p> <p>Você tem a opção de ajudar seu oponente. Se decidir ajudar ele caminhará duas casas e você uma casa, se não ajudar, vocês dois ficarão no mesmo lugar.</p>	<p>Decisão</p> <p>Você encontrou um atalho. sorteie mais um número para andar casas. Se a escolha for o número: 1, 7 e 8 ande uma casa, se não for esses números volte 1 casa</p>
<p>Decisão</p> <p>Você tem a opção de ajudar seu oponente. mas só se ele estiver atrás de você. Se você ajudar, você anda 3 casas e seu oponente ocupa seu lugar, se não ajudar, você volta 2 casas e ele fica no lugar.</p>	<p>Decisão</p> <p>Se você estiver atrás de seu oponente, troque de lugar com seu oponente. Se não tiver atrás de seu oponente, ande uma casa.</p>	<p>Decisão</p> <p>Se você estiver atrás de seu oponente, você poderá andar 3 casa e seu oponente voltará 1 casa. Se estiver na frente, fique no lugar.</p>

PLANO DE AULA

Série: 8º ano

Disciplina: Pensamento Computacional

Tema da aula: Compreendendo estruturas de repetição

Foco e objetivos da aula:

- Compreender como funciona uma estrutura condicional

Materiais necessários:

- Folha de atividade
- Papel em branco

Nexos conceituais de Navarro:

Pensamento algébrico: Identificar os padrões que gerem o looping.

Pensamento algorítmico: Identificando os passos necessários para a criação do cronômetro.

Resolução de problemas: Como montar um cronômetro.

Pilares de PC segundo Brackmann:

Abstração: Identificar onde o looping será criado.

Decomposição: Identificar os comandos que ficam fora e dentro da estrutura de repetição.

Reconhecimento de padrões: Compreender o padrão que se repete.

Algoritmo: O passo a passo.

Estrutura / Atividade:

(Tempo: 10 minutos)

1) Explicar o que é uma estrutura condicional de acordo com o subtítulo:

estruturas de repetição  informado neste ebook.

(Tempo: 40 minutos)

2) Distribuir as folhas de atividade para os estudantes e uma folha em branco. Os estudantes deverão recortar os comandos da estrutura de repetição de um cronômetro e colar montando o looping correto.



Sempre que encontrar uma seta em uma frase, isso indica que o item é clicável e levará você diretamente ao conteúdo correspondente.

Avaliação:

A avaliação será realizada por meio da entrega da atividade realizada pelos estudantes. Serão considerados os seguintes critérios:

- Avaliar se o algoritmo foi montado da maneira correta.

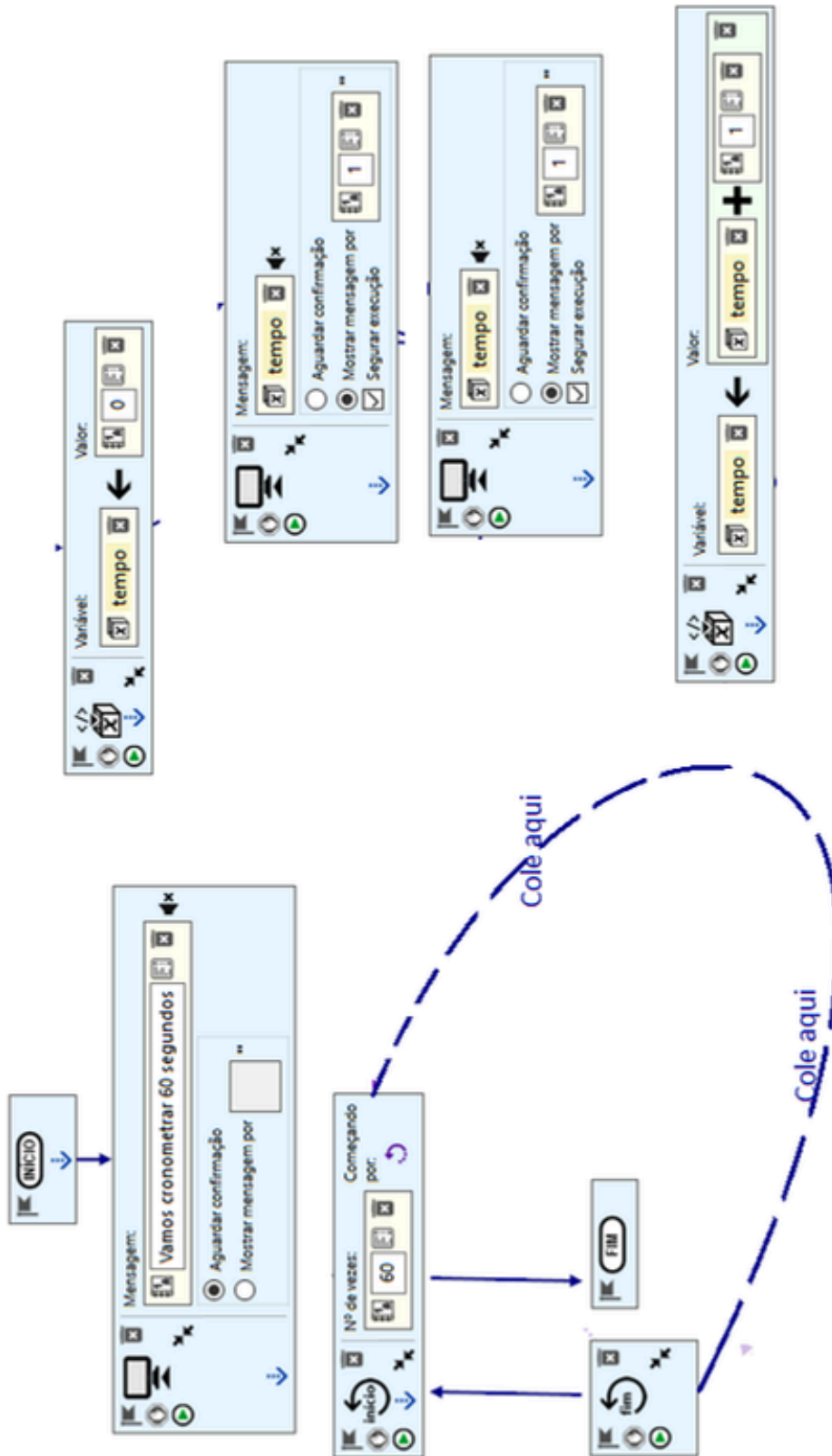
FOLHA DE ATIVIDADES

1/2

Nome do estudante:

Série:

Disciplina: Pensamento Computacional



05

**REFERENCIAL
TEÓRICO**



BRACKMANN, C. P. **Desenvolvimento do pensamento Computacional através de atividades desplugadas na educação básica.** 2017. 224 f. Tese (Doutorado em Informática na Educação) – Programa de Pós-Graduação em Informática na Educação, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017. Disponível em: <https://lume.ufrgs.br/handle/10183/172208>. Acesso em: 6 abr. 2025.

NAVARRO, E. R. **O desenvolvimento do conceito de pensamento computacional na educação matemática segundo contribuições da teoria histórico-cultural.** 2021. Tese (Doutorado em Educação) – Universidade Federal de São Carlos, São Carlos, 2021. Disponível em: <<https://repositorio.ufscar.br/handle/ufscar/15112>>. Acesso em: 28 abr. 2024.

WING, J. M. Computational Thinking. **Communications of the ACM**, n. 3, v. 49, p. 33-35. 2006. Disponível em: <https://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf> . Acesso em: 11 jun. 2024.

WING, J. M. Research notebook: computational thinking—what and why. **The Link Magazine**, v. 6, p. 20-23, 2011. Disponível em: <https://people.cs.vt.edu/~kafura/CS6604/Papers/CT-What-And-Why.pdf>. Acesso em: 27 mar. 2025.

ZATTI, E. A. **GenIA**: plataforma para construção de objetos de aprendizagem que faz uso de programação intuitiva e é assistida por inteligência artificial. 2023.121 f. Tese (Doutorado em Formação Científica, Educacional e Tecnológica) – Universidade Federal do Paraná, Curitiba, 2023.