

PARA COMEÇAR

Esta etapa aborda conceitos relacionados à **lógica de programação**.

No decorrer de seus estudos, esperamos que você desenvolva as seguintes capacidades:

- aplicar técnicas de programação na elaboração de algoritmos inerentes aos sistemas de TI;
- aplicar linguagens de programação para elaborar programas e sistemas de TI.

Para desenvolver tais capacidades, você deverá estudar os seguintes temas:

- lógica de programação e algoritmos;
- softwares;
- técnicas e linguagens de programação;

Esse estudo será necessário para que você resolva a situação-problema a seguir. Então, avance para conhecê-la.

SITUAÇÃO-PROBLEMA

A empresa Savinis, focada no desenvolvimento de software de alta performance, contratou você para realizar o desenvolvimento de um sistema de controle de estoque de peça que será disponibilizado para os funcionários da empresa contratante, o que, até então, era realizado manualmente.

Esse sistema terá como função principal a de controlar a entrada e saída de peças do estoque, além de atender aos seguintes requisitos:

- a empresa possui apenas uma peça a ser controlada;
- o sistema deverá solicitar sempre o saldo inicial da peça uma única vez a cada vez que o sistema é iniciado;
 - 1 = compra (entrada) de peças;

- 2 = venda (saída) de peças;
- enquanto o usuário não encerrar a entrada de dados, o sistema deverá continuar solicitando nova entrada de dados:
 - os dados de entradas são: tipo de entrada e quantidade de peças;
 - caso a entrada seja do tipo 1 (compra), somar a quantidade ao estoque da peça;
 - caso a entrada seja do tipo 2 (venda), subtrair a quantidade do estoque da peça caso a quantidade de entrada seja inferior ou igual ao saldo de estoque. Caso seja informado uma quantidade maior que o saldo da peça, apresentar a mensagem “Saldo insuficiente” e desconsiderar atualização de saldo;
 - a cada entrada de dados, apresentar o saldo atualizado do estoque;
 - ao final de cada entrada de dados, perguntar se o usuário deseja continuar (s) ou não (n) a entrada de dados;
- caso o usuário encerre o sistema, apresentar a mensagem "Sistema encerrado".

Para criação desse sistema de cadastro, você, enquanto programador(a) responsável, deverá solucionar os desafios a seguir.

Clique nos botões a seguir para conhecê-los.

Desafio 1



Descrever a sequência de passos lógicos necessários para criação do sistema solicitado pela empresa.

Desafio 2



Aplicar técnicas de programação e ferramentas para desenvolvimento do código.

Desafio 3



DESAFIO 1

Nesta etapa, você deverá resolver o Desafio 1:

- descrever a sequência de passos lógicos necessários para criação do sistema solicitado pela empresa.

Para isso, você estudará os seguintes conteúdos:

- lógica de programação e algoritmos;
- softwares.



O QUE É LÓGICA?

A lógica é o campo de estudo que utiliza princípios e conhecimentos para se atingir um raciocínio correto.

O filósofo Copi (1978) define lógica como:

estudo dos métodos e princípios usados para distinguir o raciocínio comum e natural ao ser humano que nem
correto do incorreto.

está trancada?

Para isso, deve-se executar a sequência a seguir.

A utilização do raciocínio lógico é tão
comum e natural ao ser humano que nem
percebemos.

Pense, por exemplo: quais são os passos
necessários para passar por uma porta que





A ação de abrir uma porta trancada parece simples, mas envolve uma sequência lógica de passos. Se você tentar girar a maçaneta, antes de destrancar a porta com a chave, o resultado não será o mesmo.

Podemos citar várias outras situações cotidianas que envolvem raciocínio lógico, como tomar banho, cozinhar e dirigir um carro.

VOCÊ SABIA?

O filósofo grego **Aristóteles** (384–322) foi quem iniciou os estudos da lógica. O conjunto de sua obra é conhecida como lógica aristotélica ou lógica clássica.

No século XIX, o filósofo alemão **Gottlob Frege** desenvolveu um método chamado cálculo de predicados, que analisa proposições linguísticas por meio de processos dedutivos matemáticos, contribuindo para criação dos códigos de programação de computadores.

LÓGICA DE PROGRAMAÇÃO

A lógica de programação surgiu a partir dos princípios da lógica e consiste em uma técnica de encadeamento do pensamento para atingir um determinado objetivo ou solucionar um problema.

Podemos dizer que a lógica é a primeira etapa da programação em si. Isso quer dizer que, antes de começar a escrever o código, você deve pensar quais são questões que devem ser resolvidas, estudar quais serão as soluções possíveis e planejar todas as etapas da solução.

Para isso, é necessário utilizar uma sequência lógica, que é um conjunto de passos a serem executados.

Quando falamos em sequência lógica, estamos falando de **algoritmos**.

Saiba que uma das ações mais comuns de um(a) programador(a) é buscar as melhores soluções e encontrar os algoritmos mais adequados para a criação de programas de computadores, soluções e serviços.


Complicou?

Não se preocupe! Clique em play e assista ao vídeo a seguir para entender como tudo isso funciona.

SOFTWARES

Você viu que um software é um agrupamento de programação que são lidos pelo computador.

Em outras palavras, software é um produto de scripts, macros e instruções de código em uma máquina deve fazer.



Software embarcado ou *firmware* é um conjunto de instruções operacionais programadas pelo fabricante diretamente no hardware do equipamento, para manter a configuração básica das funções.

Isso significa que os códigos transcritos por esse tipo de programa são fundamentais para iniciar e executar os hardwares e seus recursos, fornecendo informações idênticas sempre que o dispositivo for ligado.

E você sabe dizer qual é a diferença entre **hardware** e **software**?

Clique no botão abaixo para descobrir.



TIPOS DE SOFTWARES

Os softwares são divididos em três principais categorias, são elas:

Programação	Sistema	Aplicação

São softwares que, a partir de linguagens de programação, como Java, Python e Swift, são utilizados para o desenvolvimento de outros programas.

APLICATIVOS

Aplicativos, ou apps, são softwares presentes em dispositivos móveis, como smartphones, tablets, smart TVs, relógios inteligentes, entre outros, que desempenham vários tipos de tarefas.

Apresentam interface amigável e simples para facilitar a interação com o usuário. Alguns já vêm instalados e outros podem ser adquiridos em lojas de aplicativos e podem, ainda, apresentar versões gratuitas ou pagas.

WhatsApp, Uber e Ifood são alguns exemplos desses aplicativos.



VOCÊ SABIA?

Qual é a diferença entre um celular e um smartphone?

O smartphone possui sistema operacional, como os computadores, e o celular comum não possui. O celular realiza tarefas mais simples, como digitar e enviar mensagens. Já o smartphone, além realizar essas tarefas, possui várias outras funções, como acesso à internet, jogos, loja de aplicativos etc.

| TIPOS DE APLICATIVOS

Dependendo de como serão utilizados, o desenvolvimento de um aplicativo requer recursos e tecnologias específicas.

Atualmente, existem basicamente três tipos de aplicativos móveis: **nativos**, **web** e **híbridos**.

Clique nas abas para conhecer cada um deles.

Nativos
Web apps
Híbridos

SISTEMAS OPERACIONAIS

Um sistema operacional, também chamado de SO, é um tipo de software de sistema que controla praticamente todos os processos de um computador.

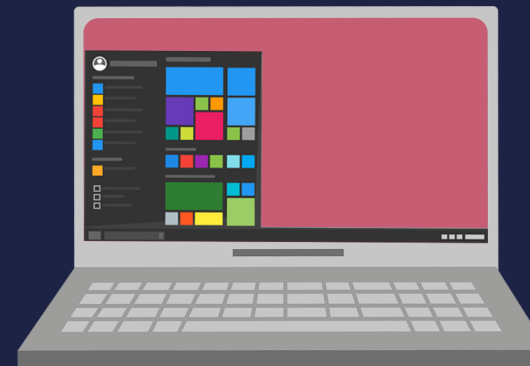
Em outras palavras, sistema operacional é o software responsável por fazer a ponte, a interface entre o usuário e o hardware, gerenciando recursos do sistema e do hardware.

Algumas funções básicas dos sistemas operacionais são:

- controlar acesso ao hardware;
- prover interface ao usuário;
- gerenciar programas;
- gerenciar arquivos e pastas.

O usuário interage com a interface do SO, mas não precisa gerenciar os recursos.

Podem-se destacar três estruturas fundamentais para o funcionamento adequado do computador: **shell**, **kernell** e **hardware**.



Clique na imagem para conhecer cada uma delas.



| CARACTERÍSTICAS ENTRE AS INTERFACES GUI E CLI

Clique nas abas laterais para ver as características e diferenças entre a interface GUI (gráfica) e a CLI (linha de comando).

GUI

Características da interface gráfica do usuário:

- exibição de tela em um monitor;
- utilização do mouse para seleção e execução de programas;

1/2

keyboard_arrow_right

- utilização do teclado para inserção de dados.



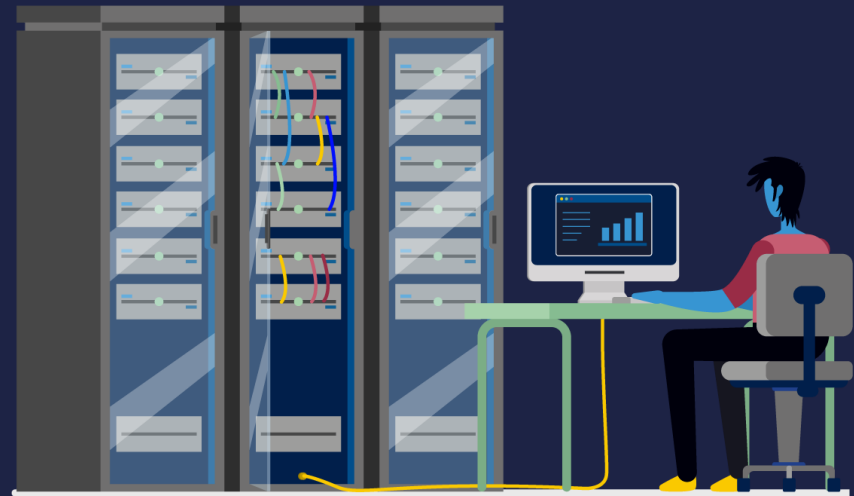
CATEGORIAS DE SO

O sistema operacional pode ser classificado em duas categorias: para desktops e para servidores.

Clique nas abas para conhecer cada um deles.

Sistemas operacionais para desktops

Sistemas operacionais para servidores



Os sistemas operacionais para desktops mais comuns são: Windows (Microsoft), MacOS (Apple) e Linux (de código aberto).

Clique nas abas a seguir para conhecer detalhes de cada um deles.

Windows (Microsoft)

MacOS (Apple)

Linux

DRIVERS

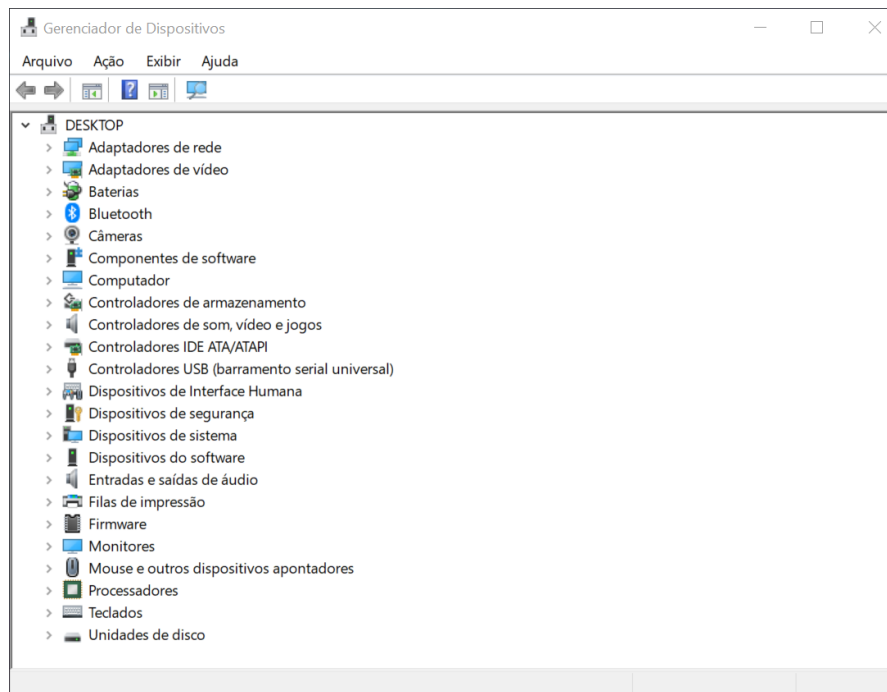
Os drivers são softwares ou pequenos elementos de softwares que ligam o SO aos dispositivos físicos do computador.

Os drivers são responsáveis pela comunicação entre os sistemas operacionais e um componente de hardware.

Então, se você conectar um dispositivo, como uma impressora ou uma placa de vídeo, e não instalar o driver apropriado, o computador pode não reconhecer aquele componente que foi conectado, e não poderá utilizá-lo.

Você pode acessar os drivers de seu computador por meio do gerenciador de dispositivos. A imagem a seguir mostra o gerenciador de dispositivos de um SO Windows.

Clique para ampliá-la.



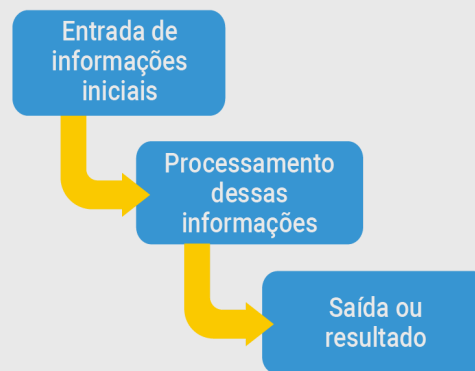
IMPORTANTE

Compreender as diferenças entre hardware, software e driver, bem como os conceitos apresentados, irão permitir a você, programador ou programadora, o melhor gerenciamento dos recursos e da manipulação dos itens necessários para que você desenvolva seu projeto. Isso fará com que você compreenda, ao desenvolver um código lógico, como o seu computador estará processando e compartilhando essa informação.

Ao desenvolver um projeto prático, utilizar recursos de maneira errada pode prejudicar a performance do seu software e do seu hardware.

ALGORITMOS

Como já mencionamos, desenvolvimento de qualquer software é realizado por meio dos algoritmos, os quais descrevem uma sequência de passos lógicos necessários para a execução de uma tarefa e que consistem em:



É importante salientar que uma tarefa pode ser composta de várias pequenas ações e cada uma delas tem seu próprio conjunto de instruções ou algoritmo a ser seguido.

Para exemplificar, considere o o problema a seguir.

NOTAS

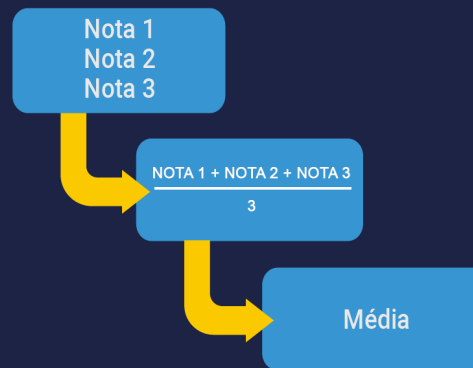
Você precisa criar um algoritmo que, a partir da soma de três notas semestrais de um aluno, calcule a média final dele para, na sequência, informar se o aluno foi reprovado ou aprovado, considerando as seguintes referências:

- de 7 a 10 – aprovado;
- abaixo de 7 – reprovado.

Para resolução desse problema, serão necessários dois algoritmos, sendo que cada um resolverá um pequeno problema da situação.

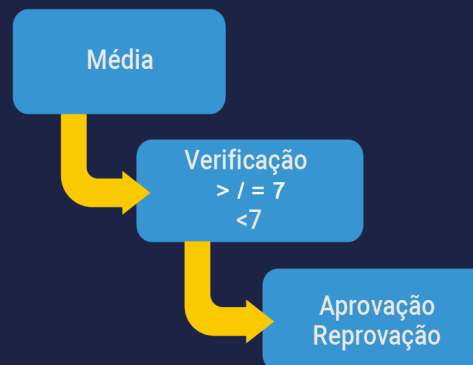
1º ALGORITMO

Calculará a média do aluno. Ele deve possuir como entrada as três notas obtidas durante o semestre. O processamento deverá realizar o cálculo da média, somando as notas e dividindo o resultado por três. Por fim, a saída (ou resultado) será a média obtida pelo aluno.



2º ALGORITMO

Verificará a situação final do aluno. Para isso, a entrada desse algoritmo será a média obtida com o primeiro algoritmo. O processamento será a verificação dessa média, para descobrir se foi maior, igual ou menor que 7. Com essa verificação, é possível definir a saída deste algoritmo, informando se o aluno foi aprovado ou não.



E COMO REALIZAR A REPRESENTAÇÃO DOS ALGORITMOS?

A representação dos algoritmos pode ser realizada por meio de **descrição narrativa**, **fluxograma** e **pseudocódigos**.

Siga em frente para conhecer cada uma delas.

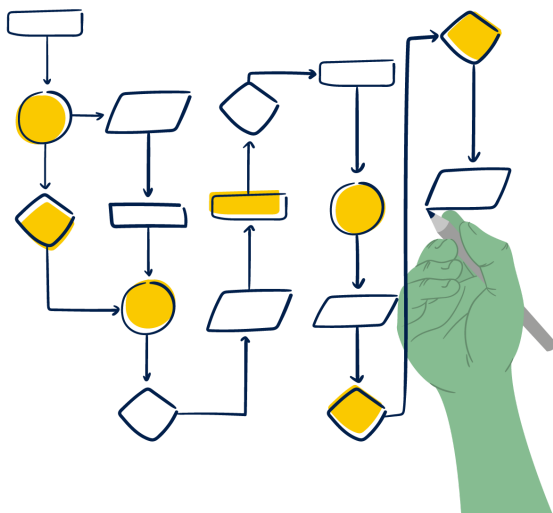
DESCRIÇÃO NARRATIVA

Utiliza palavras para expressar os algoritmos e, por isso, é exclusivamente usada para fins didáticos. O algoritmo apresentado para fazer café é um exemplo desse tipo de representação.

O exemplo abaixo mostra o algoritmo do problema sobre notas, representado por meio da descrição narrativa.

1. Obter as 3 notas do aluno.
2. Somar as 3 notas.
3. Dividir o resultado por 3.
4. Se a média for maior ou igual a 7 = aluno foi aprovado.
5. Se a média for menor que 7 = aluno foi reprovado.





FLUXOGRAMA

Fluxograma é um tipo de diagrama ou uma representação gráfica que descreve as diferentes ações a serem realizadas durante a execução de um algoritmo.

Ele auxilia na elaboração do raciocínio lógico a ser seguido para a resolução de um problema, mostrando visualmente como o nosso código deve se comportar nas diversas situações e as diferentes saídas que ele terá dentro do nosso programa.

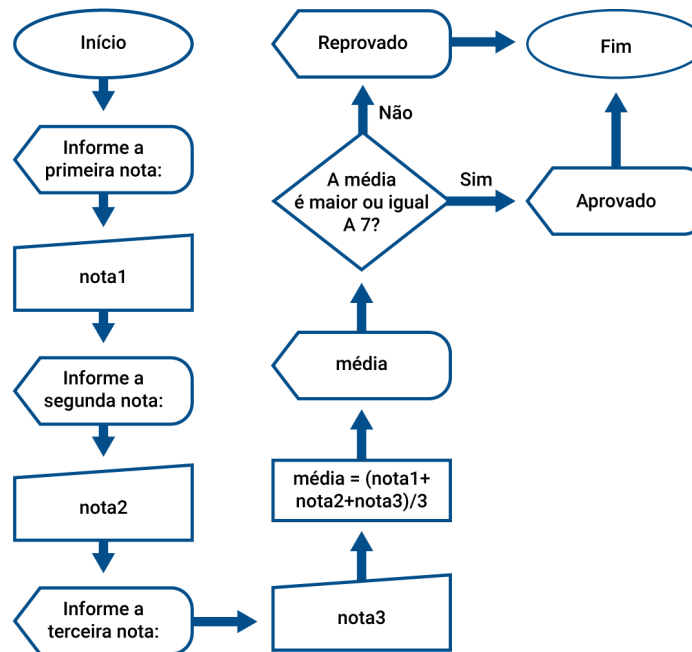
Por exemplo: não temos a água para realizar o café, o que fazer?

Dessa forma, usando as linguagens de programação, escreveremos ações para que nosso código tenha o comportamento específico, exatamente como exibido no fluxograma, e seja concluído com sucesso.

A tabela mostra as simbologias mais comuns utilizadas para representação de algoritmos por meio de fluxogramas:

O fluxograma a seguir mostra a representação do algoritmo do problema sobre notas.

Representação gráfica	Item	Descrição
	Início/fim	Todo fluxograma deve iniciar e encerrar com este símbolo. O fluxograma deve conter apenas um início; porém, poderá possuir mais de um fim, pois pode se dividir durante o processo.
	Leitura	Representa uma entrada do usuário, quando o programa fará uma leitura de uma informação digitada por ele.
	Escrita	Representa a impressão de alguma informação na tela pelo programa com o objetivo de informar o usuário.
	Seta de fluxo	Representa o caminho do fluxograma. A partir do início, a leitura do fluxograma é feita seguindo as setas do fluxo.
	Processo	Utilizado quando algo deve ser processado pelo programa, por exemplo, para cálculos matemáticos.
	Decisão	Divide a execução do fluxograma em dois caminhos. Sempre que for utilizado, uma pergunta deve ser feita. Caso a resposta seja verdadeira, o fluxograma segue por um caminho, caso contrário, segue por outro. Essa é a única situação em que devem ser utilizadas duas setas de fluxo a partir de uma figura geométrica.



Perceba que o fluxograma mostra o passo a passo de cada ação. Primeiro, solicita-se as 3 notas, utilizando um nome genérico, como notas 1, 2 e 3. Depois, é realizado um processo para calcular a média [$\text{média} = (\text{nota } 1 + \text{nota } 2 + \text{nota } 3) / 3$], cujo resultado fornece a média do aluno.

Além de calcular a média, o fluxograma também apresentará se o aluno foi reprovado ou aprovado, considerando a média 7 para aprovação.

TESTANDO NA PRÁTICA

Para testar a execução e a eficácia do fluxograma, simule o passo a passo e identifique se o resultado final está correto ou não, utilizando os seguintes valores de notas:

1. Nota 1: 7
2. Nota 2: 8
3. Nota 3: 8

DICA

Há vários programas gratuitos na internet, com opções online e offline para o desenvolvimento de fluxogramas. Conheça alguns:

[LucidChart](#)

[Draw.io](#)

E então? Qual foi a média desse aluno? Ele foi aprovado ou reprovado?

Clique no botão abaixo para descobrir.

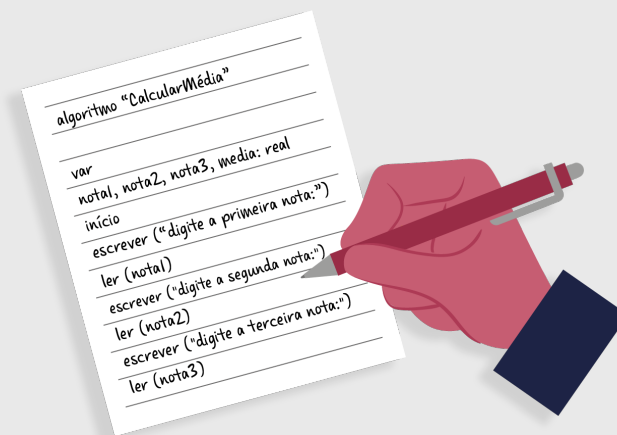


PSEUDOCÓDIGOS

É uma forma de representação de algoritmo semelhante à linguagem de programação, porém utilizando palavras no idioma escolhido.

O pseudocódigo apresenta 3 etapas:

1. identificação do algoritmo;
2. declaração das variantes;
3. corpo do algoritmo.



A seguir, acompanhe o exemplo do cálculo da média representado pelo pseudocódigo.

```
1. algoritmo "CalcularMédia"
2. var
3. nota1, nota2, nota3, media: real
4. inicio
5. escrever ("digite a primeira nota:")
6. ler (nota1)
7. escrever ("digite a segunda nota:")
8. ler (nota2)
9. escrever ("digite a terceira nota:")
10. ler (nota3)
11. media<-(nota1+nota2+nota3) /3
12. escrever (media)
13. se media>=7 então
14. escrever ("Aprovado")
15. senão
16. escrever ("Reprovado")
```

Linha 1: identificação do algoritmo.

Linhas 2 e 3: declaração das variantes, ou seja, notas 1, 2 e 3.

Linhas 4 a 18: corpo do algoritmo: notas, cálculo da média, decisão e verificação.

ENTENDENDO ALGUNS TERMOS

Você percebeu que foram utilizados alguns termos e símbolos específicos para representação dos algoritmos em pseudocódigos.

Clique nas abas a seguir para conhecer cada um deles.

17. fimse
18. fim

Algoritmo
Var
Início
Escrever
Ler
Fim



LINGUAGEM DE PROGRAMAÇÃO

É um método padronizado, formado por um conjunto de regras sintáticas e semânticas, símbolos e códigos que torna possível a execução do algoritmo.

É por meio dessa linguagem de programação que o programador se comunica com a máquina, faz um site funcionar e desenvolve softwares, aplicativos para sistemas operacionais, jogos etc.

Existem vários tipos de linguagem de programação e você estudará mais sobre elas no Desafio 3, porém, antes, é necessário que você conheça algumas técnicas de programação comuns em todas as linguagens.

Por ora, conheça como o exemplo do cálculo da média ficou após implementado na linguagem Dart.

```
1  import 'dart:io';
2  import 'package:intl/intl.dart';
3
4  final fmt = NumberFormat('#,##0.00', 'pt_BR');
5
6  void main(List<String> args) {
7      stdout.write('digite a primeira nota: ');
8      var n1 = fmt.parse(stdin.readLineSync());
9      stdout.write('digite a segunda nota: ');
10     var n2 = fmt.parse(stdin.readLineSync());
11     stdout.write('digite a terceira nota: ');
12     var n3 = fmt.parse(stdin.readLineSync());
13
14     var media = (n1 + n2 + n3) / 3;
15
16     if(media >= 7) {
17         print('Aprovado');
18     } else {
19         print('Reprovado');
20     }
21 }
```

VAMOS PRATICAR

1. Marque verdadeiro ou falso para as afirmações a seguir.

	Verdadeiro	Falso
a. O raciocínio lógico só é usado na programação.	<input type="checkbox"/>	<input type="checkbox"/>
b. No algoritmo, o importante é completar as etapas, não importa a sequência.	<input type="checkbox"/>	<input type="checkbox"/>
c. A lógica é a primeira etapa da programação.	<input type="checkbox"/>	<input type="checkbox"/>
d. Algoritmos são usados no dia a dia.	<input type="checkbox"/>	<input type="checkbox"/>
e. Só existe um algoritmo correto para cada tarefa complexa.	<input type="checkbox"/>	<input type="checkbox"/>

VERIFICAR RESPOSTA

2. Software é um produto virtual usado para descrever programas, aplicativos, scripts, macros e instruções de código embarcado diretamente (*firmware*), a fim de determinar o que uma máquina deve fazer.

Relacione corretamente as definições a seguir e seu respectivo tipo de software ou app.

Realiza ações específicas solicitadas pelo usuário.	<div>Selecione</div>
Permite a execução de outros softwares.	<div>Selecione</div>
É utilizado para o desenvolvimento de outros programas.	<div>Selecione</div>
É desenvolvido para ser aberto no navegador do smartphone.	<div>Selecione</div>

Identifique a alternativa correta:

a. ☐ Somente a sequência

Pacote de software que consolida as ferramentas básicas necessárias para escrever e testar software.

Selecione ▼

VERIFICAR RESPOSTA

3. O sistema operacional é o software responsável por fazer a ponte, a interface entre o usuário e o hardware.

Associe corretamente as estruturas de funcionamento do computador de acordo com suas funções.

Interface de interação com o usuário.

Selecione ▼

Interface de interação entre hardware e software.

Selecione ▼

Componentes físicos.

Selecione ▼

Interface gráfica.

Selecione ▼

Interface de linha de comando.

Selecione ▼

VERIFICAR RESPOSTA

4. Um algoritmo descreve uma sequência de passos lógicos necessários para a execução de uma tarefa.

Considere as sequências narrativas, abaixo, de um algoritmo que descreve uma receita de bolo:

Sequência 1

1. Quebre os ovos.
2. Misture os ovos com o leite e a farinha.
3. Unte a forma.
4. Despeje a mistura na forma.
5. Ligue o forno.
6. Leve a forma ao forno.

Sequência 2

1. Ligue o forno.
2. Unte a forma.
3. Quebre os ovos.
4. Misture os ovos com o leite e a farinha.
5. Despeje a mistura na forma.
6. Leve a forma ao forno.

VERIFICAR RESPOSTA

5. O fluxograma a seguir descreve um algoritmo para verificação de uma senha. Aplique os valores abaixo, simule o passo a passo e identifique a alternativa que traz o resultado final correto.

Senha A = 18

Senha B = 15

Clique na imagem para ampliá-la.

VERIFICAR RESPOSTA

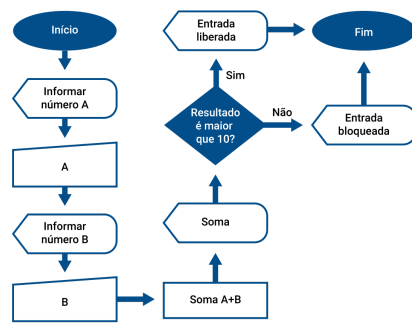
1 está
correta.

b. ☐ Somente
a sequência
2 está
correta.

c. ☐ A
sequência 1
está
parcialmente
correta.

d. ☐ A
sequência 2
está
parcialmente
correta.

e. ☐ As duas
sequências
estão
corretas.



- a. ☐ 34 - entrada liberada
- b. ☐ 33 - entrada liberada
- c. ☐ 03 - entrada bloqueada
- d. ☐ 09 - entrada bloqueada

6. No exercício anterior, você simulou um algoritmo para verificação de senha representado por um fluxograma. Considere, agora, a representação desse mesmo algoritmo em pseudocódigo e identifique a alternativa correta.

```

1. algoritmo "VerificarSenha"
2. númeroA, númeroB, soma : real
3. início
4. escrever ("digite número A:")
5. ler (A)
6. escrever ("digite número B:")
7. ler (B)
8. Soma>(A+B)
9. escrever (Soma)
10. se soma>10 então
11. escrever ("Entrada Liberada")
12. senão
13. escrever ("Entrada Bloqueada")
14. fimse
15. fim
  
```

- a. ☐ A descrição do algoritmo está completa.
- b. ☐ A descrição do algoritmo está incompleta, pois está faltando o comando que informa o início do programa.
- c. ☐ A descrição do algoritmo está incompleta, pois está faltando o comando "escrever".
- d. ☐ A descrição do algoritmo está incompleta, pois está faltando o comando "Var"
- e. ☐ A descrição do algoritmo está incompleta, pois está faltando o comando "ler".

VERIFICAR RESPOSTA

NESTE DESAFIO...

LÓGICA DE PROGRAMAÇÃO | DESAFIO 1



Você percebeu que a lógica de programação usa o raciocínio lógico de forma estruturada e planejada na hora de escrever os códigos, por meio de algoritmos.

Algoritmo é a base de todo o sistema. Consiste em um conjunto de tarefas que devem ser realizadas de maneira sequencial e planejada para resolver algum problema, criar produtos e serviços. Ele pode ser representado por descrição narrativa, fluxograma, pseudocódigos e é executado por meio das linguagens de programação. Todos os tipos de softwares são basicamente algoritmos.

Falando em softwares, você estudou o como é importante entender alguns conceitos básicos, como hardwares, aplicativos, sistemas operacionais e drivers, para utilizar os recursos de maneira assertiva.

NO PRÓXIMO DESAFIO...

Vamos conhecer outros tópicos importantes dentro das técnicas de programação.