Programar não é digitar códigos, é resolver problemas!

-O que é lógica?

● Coerência de raciocínio, de ideias;

● Sequencia coerente, regular e necessária de acontecimentos, coisas;

● Lógica de programação significa contextualizar a lógica na progr de pcs, buscando a melhor sequencia de ações (algoritmo) pra solucionar um problema.

-Abstração: se concentrar nos aspectos essenciais, ignorando características menos importantes ou acidentais.

-Algoritmo: sequencia de passos que resolve um problema (em programação, SEMPRE deve ter mostrado inicio e fim do código).

-Pseudocódigo: forma genérica de escrever um algoritmo usando linguagem simples (linguagem nativa, português, todos conseguem entender mesmo sem saber programação).

Exercícios finais:

1. Criar um algoritmo do seu dia
2. Tentar resolver todos os exercícios do [*www.studio.code.org/s/mc/stage/1/puzzle/1*](http://www.studio.code.org/s/mc/stage/1/puzzle/1)em menos de 1h e se conseguir, marcar @digitalinnovation.one e @denilsonbonatti no insta
3. Fazer exercício do *rachacuca.com.br/jogos/pinguins-numa-fria/* e escrever um algoritmo de como resolve-lo.

-Fluxograma: ferramenta usada para representar graficamente o algoritmo, isto é, a sequência lógica e coerente do fluxo de dados. Faz a documentação dos passos necessários para a execução de um processo.

-Variáveis: na programação, é um objeto (uma posição, frequentemente localizada na memoria) capaz de reter e representar um valor ou expressão. É um espaço na memoria do pc destinado a um dado que é alterado durante a execução do algoritmo. São classificadas em:

● Numéricas (inteiras, decimais, reais);

● Caracteres;

● Alfanuméricas (quando tem letra);

● Lógicas (verdadeira ou falsa).

-Constante: valores imutáveis que não são alterados durante a vida útil do programa (ex: pi=3,14 sempre, nunca muda).

\*Tomadas de decisão e expressões

-Expressões aritméticas: utilizam operadores aritméticos (+ , - , \* , / , ^ , % , etc) e funções aritméticas envolvendo constantes e variáveis (ex: 50 + 50 = 100).

-Expressões literais: usam constantes e/ou variáveis que tem como resultado valores literais (ex: nome=“José Silva”, igual em Java).

-Operadores relacionais: expressões compostas por outras expressões ou variáveis numéricas com operadores relacionais (> , >= , < , <= , == , != ). Sempre retornam valores lógicos (verdadeiro/falso).

-Tomadas de decisão: 60% da programação. Necessidade de decidir o que fazer dependendo de alguma condição encontrada durante a execução.

\*Concatenação: termo usado em computação para designar a união do conteúdo de duas strings (sequencia de caracteres). Em Java é usado +, em fluxograma é usado & (ex: “O seu nome é: “ & nome & “e o seu sobrenome é: “ & sobrenome)

**INTRODUÇÃO AO PORTUGOL**

\*Estrutura de repetição: estrutura que permite executar mais de uma vez o mesmo comando ou conjunto de comandos, de acordo com uma condição com um contador.

\*Linguagens de programação:

-O que são as linguagens de programação?

Linguagem escrita e formal que especifica um conjunto de instruções e regras usadas pra gerar programas (software). Um software pode ser desenvolvido pra rodar em pc, dispositivo móvel ou em qualquer equipamento que permita sua execução. A função das linguagens é servir de um meio de comunicação entre pcs e humanos.

-Separadas em:

● alto nível: aquelas cuja sintaxe se aproxima mais da linguagem humana;

● baixo nível: a que se aproxima mais da linguagem da máquina, precisa ter o conhecimento direto da arquitetura do pc pra fazer alguma coisa.

● compiladas: o código fonte é executado diretamente pelo sistema operacional ou pelo processador após ser traduzido por um processo chamado compilação;

● interpretadas: o código fonte é executado por um programa de pc chamado interpretador que é executado pelo sist. Oper ou processador.

-O que é Portugol?

Pseudolinguagem que permite ao leitor desenvolver algoritmos estruturados em português de formas simples/intuitiva, independente da linguagem de program. Aprende a resolver o problema sem pensar se é um app, um site, etc.

-No Portugol são utilizados outros “nomes” de variáveis para criar funções estilo Java:  
● *real nota1,nota2* (quando vc vai ter um numero real como resultado);

● *cadeia aluno* (quando vc quer uma string, uma palavra, nomes, etc);

● *caracter* (quando são variáveis de A-Z apenas);

● *escreva(“Nome do aluno”)* (comando que imprime o que está escrito);

● *leia(aluno)* (vai entender que o que for escrito nesse local é a variável aluno).

Exercicio final:

1. Criar um código de vendas de um funcionário de janeiro a abril exibindo o total e a média. Marcar Denilson Bonatti no instagram.

\*Desvios condicionais

-**SE***:* é utilizada a palavra reservada (utilizada pra uma determinada função no código, não pode ter variável com esse nome) SE para dizer uma condição, a condição a ser testada deve estar entre parênteses e as instruções devem ser executadas entre chaves caso o desvio seja **verdadeiro**.

Ex: se(media>=7) {

Escreva(“Parabens, vc foi aprovado!”)

}

-**SE-SENAO:** se a condição for **falsa**, usa-se o senão.

Ex: se(media>=7) {

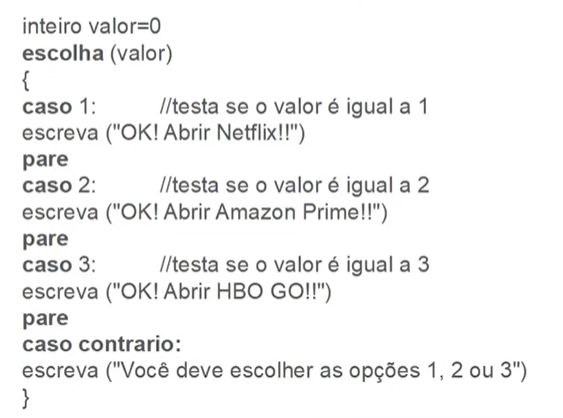
Escreva(“\n” + “Parabens, vc foi aprovado!”)

}

Senao {

Escreva(“\n” + “Infelizmente vc foi reprovado”)  
}

-**CASO**: similar ao se e senão, e reduz a complexidade na escolha de diversas opções. Neste comendo não é possível o uso de operadores lógicos, apenas trabalha com valores definidos. Ex:



\*Comentários

-Feitos para que outros programadores (da mesma empresa, por ex) entendam o motivo de cada parte do seu código. Normalmente são feitos no inicio do código ou antes de alguma parte que vc queira explicar especificamente. Ex:

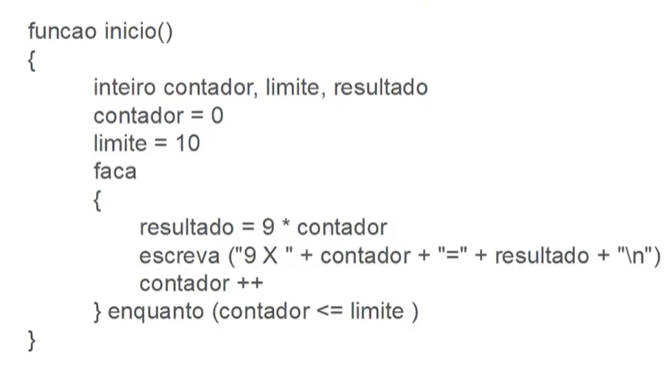
//Função do algoritmo: Calcular a média aritmética

//Autora: Beatriz Wenzel

//Essa parte verifica se a média do aluno é maior ou igual a 7

\*Laços de repetição no Portugol

-Estrutura que permite executar mais de uma vez o mesmo comando ou conjunto de comandos, de acordo com uma condição com um contador. (obs: *contador++* **=** *contador=contador+1*)

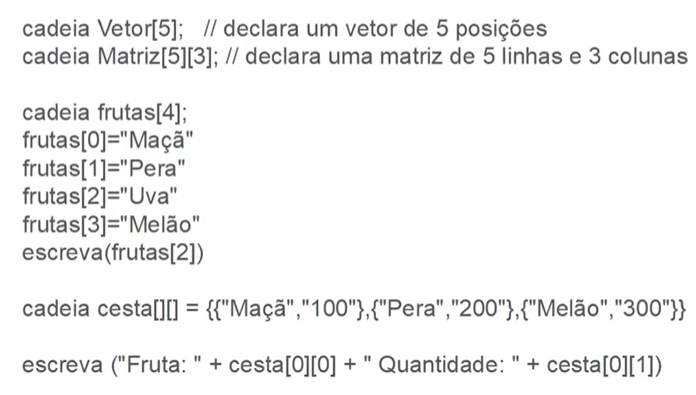


-Faça: vai reproduzir o que está dentro dos {} mas o **enquanto** faz com que o **faça** se repita, pois o contador++ sempre add 1 no contador, então enquanto não chegar no limite=10 ele continua repetindo).

\*Matrizes e vetores

-Matriz: coleção de variáveis do mesmo tipo, acessíveis com um único nome e armazenados em sequencia na memória. A individualização de cada variável de um vetor é feita através do uso de índices;

-Vetor: matrizes de uma só dimensão (ex: vetor de 5 colunas ou de 4 linhas).





|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| 0 | Pera | 100 |
| 1 | Jaca | 200 |
| 2 | Maçã | 900 |
| 3 | Uva | 89 |

Cesta[0][0] = Pera

Cesta[0][1] = 100

Como a função exibe “cesta[**contador**][0]” vai começar com contador=0 mas o **contador++**  aumenta esse número em 1 na próxima repetição, fazendo com que se crie uma lista como a tabela acima, com os respectivos nomes das frutas e suas quantidades.