**SESC ESCOLA CUIABÁ**

**Marcos Vinicius Rolins Mendes**

**ALGORITMOS E LÓGICA DE PROGRAMAÇÃO**

**CUIABÁ**

**2024**

**Marcos Vinicius Rolins Mendes**

**ALGORITIMOS E LÓGICA DE PROGRAMAÇÃO**

**CUIABÁS**

**2024**

**1.Algoritmo**--------------------------------------------------

1.Definição de algoritmos e sua importância na**--------------** programação

2.Lógica de programação e sua relação com algoritmos**--**

**2.Estruturas de controle--------------------------------**

1.Estruturas de controle como sequência, seleção**----------**

2.Exemplos de algoritmos que utilizam estruturas de**-------** controle

**3.Tipos de dados variáveis----------------------------**

1.Definição de tipos de dados básicos---------------

2.Exemplos de declaração de variáveis----------------------

**4. Função e modularização**

1.Conceito de função da programação

2.Lmportância da modularização na escrita de algoritmos

**Algoritmo**

**Definição de algoritmos e sua importância na programação**

Algoritmo é um conjunto de instruções, como uma receita de bolo, constituído de um número finito de passos. Para que os dados sejam processados, há a necessidade de se utilizar uma linguagem de programação, pois os computadores não executam diretamente os algoritmos.

**Lógica de programação e sua relação com algoritmos**

Enquanto a lógica de programação é a estruturação de conceitos e regras que orientam a execução do algoritmo, o algoritmo é a receita, ou seja, os comandos em um passo a passo claro e lógico que conduzem as ações para a resolução de um problema ou para atingir determinado objetivo.

**Estruturas de controle**

**Estruturas de controle como sequência, seleção**

As estruturas de controle de fluxo permitirão desenvolver algoritmos capazes de testar expressões e, a partir delas, seguir pelas instruções de forma não linear. Seleção: se (if), escolha (switch). Repetição: enquanto (while), repita (repeat), faça (do), para (for).

**Exemplos de algoritmos que utilizam estruturas de controle**

As estruturas de controle de um algoritmo são elas: seqüencial, condicional e de repetição.

**Tipos de dados variáveis**

**Definição de tipos de dados básicos**

Os tipos de dados representam as diferentes categorias de valores que podem ser armazenados em uma variável ou constante. Alguns dos tipos de dados mais comuns incluem inteiros, ponto flutuante, booleanos e strings.

**Exemplos de declaração de variaveis**

Devem ser formados por combinações de letras maiúsculas (A-Z), minúsculas (a-z), dígitos (0-9) e o caractere sublinhado ( \_ ). Não podem começar com dígitos. Não podem conter acentuações. Não podem ser uma palavra reservada da linguagem de programação (uma variável não pode se chamar por exemplo printf)

**Função e modularização**

**Conceito de função da programação**

Uma função é um bloco de código que executa alguma operação. Opcionalmente, uma função pode definir parâmetros de entrada que permitem que os chamadores passem argumentos para a função. Uma função também pode retornar um valor como saída.

**Importância da modularização na escrita de algoritmos**

Além da fácil manutenção e organização, através do princípio da modularidade, conseguimos compreender melhor o código, tornando-o mais legível, e reaproveitá-lo em diferentes partes da nossa aplicação. Dessa forma, facilitamos o trabalho em equipe e seguimos os princípios de boas práticas da programação.

**Exemplos de funções simples e sua utilização em algoritmos mais complexos**

Alguns tipos de funções:

-Função Polinomial do 1º grau ou Função Afim.

-Função Polinomial do 2º grau ou Função Quadrática.

-Função modular.

-Função exponencial.

-Função logarítmica.

-Função Polinomial do 1º grau ou Função Afim.

-Função Polinomial do 2º grau ou Função Quadrática.

-Função modular.

-Função exponencial.

-Função logarítmica.