

SSC0800 - ICC1 – Teórica

Introdução à Ciência da Computação I

**Resolução de problemas e
desenvolvimento de
algoritmos**

Prof. Cláudio Fabiano Motta Toledo

Sumário

- Análise e solução de problemas
 - Representação
 - Documentação
-

Análise e solução de problemas

- ❖ Um algoritmo é um conjunto ordenado de passos executáveis não ambíguos, definindo um processo que tem término.
 - ❖ O algoritmo é abstrato e distinto de suas representações.
 - ❖ Um único algoritmo pode ser representado de diversas formas. Por exemplo:
 - ❑ $F = (9/5)C + 32$.
 - ❑ Multiplicar a temperatura, lida em graus Celsius, por $9/5$, e então somar 32 ao produto assim obtido.
 - ❖ Algoritmo x Programa x Processo.
 - ❑ Um programa é uma das possíveis representações de um algoritmo.
 - ❑ Processo é a atividade de executar um programa e, conseqüentemente, também é a atividade de executar um algoritmo.
-

Análise e solução de problemas

- ❖ Um algoritmo é um conjunto ordenado de passos.
 - ❑ Um algoritmo deve ter uma estrutura bem estabelecida, quanto à ordem em que seus passos são executados.
 - ❑ Não significa necessariamente execução em uma seqüência preestabelecida, onde o primeiro passo é seguido por um segundo, e assim por diante.
 - ❖ Os passos de um algoritmo não podem ser **ambíguos**.
 - ❑ As instruções devem ser claras o bastante para determinar de forma única e completa as ações necessária em cada passo do algoritmo.
 - Passo1:** Converta a leitura da temperatura de graus Celsius para Fahrenheit.
 - ❑ A instrução pode ser clara para um especialista, mas ambígua para um leigo.
 - ❑ O problema reside na representação, mas não no algoritmo.
 - ❖ Algoritmos devem sempre levar o processamento a um estado de término.
-

Análise e solução de problemas

❖ Delineamento de algoritmos

- ❑ Descobrir um método para solucionar um problema.
- ❑ É o passo que mais desafios apresenta no processo de desenvolvimento de softwares.

❖ O matemático G. Polya apresenta fases para o processo de resolução de problemas.

Fase 1. Entender o problema

Fase 2. Construir um plano para solucionar o problema.

Fase 3. Colocar o plano em funcionamento.

Fase 4. Avaliar a solução quanto à precisão e quanto ao seu potencial como ferramenta para solucionar outros problemas.

Análise e solução de problemas

❖ Traduzindo para o contexto do desenvolvimento de programas:

Fase 1. Compreender o problema.

Fase 2. Adquirir uma idéia da forma como um procedimento algorítmico poderia resolver o problema.

Fase 3. Formular o algoritmo e representá-lo na forma de um programa.

Fase 4. Avaliar o programa quanto à precisão e quanto ao seu potencial como ferramenta para resolver outros problemas.

Representação e documentação

- ❖ **Descrição narrativa:** consiste em analisar o enunciado do problema e escrever, utilizando linguagem natural, os passos a serem seguidos para sua resolução.
 - ❑ Não é necessário aprender novos conceitos, pois a língua natural já é bem conhecida.
 - ❑ A língua natural abre espaço para várias interpretações, dificultando a transcrição desse algoritmo para programa.
 - ❖ **Fluxograma:** consiste em analisar o enunciado do problema e escrever, utilizando símbolos gráficos, os passos a serem seguidos para sua resolução.
 - ❑ O entendimento de elementos gráficos é mais simples que o entendimento de textos.
 - ❑ Os fluxogramas devem ser entendidos e o algoritmo resultante não é detalhado. Isso dificulta sua transcrição para um programa
-

TABELA 1.1: Conjunto de símbolos utilizados no fluxograma.

	Símbolo utilizado para indicar o início e o fim do algoritmo.
	Permite indicar o sentido do fluxo de dados. Serve exclusivamente para conectar os símbolos ou blocos existentes.
	Símbolo utilizado para indicar cálculos e atribuições de valores.
	Símbolo utilizado para representar a entrada de dados.
	Símbolo utilizado para representar a saída de dados.
	Símbolo que indica que deve ser tomada uma decisão, indicando a possibilidade de desvios.

Representação e documentação

- ❖ **Pseudocódigo:** consiste em analisar o enunciado do problema e escrever, por meio de regras predefinidas, os passos a serem seguidos para sua resolução.
 - ❑ A passagem do algoritmo para qualquer linguagem de programação é quase imediata.
 - ❑ As regras do pseudocódigo devem ser aprendidas.
-

Representação e documentação

EXEMPLO: Algoritmo para exibir o resultado da multiplicação de dois números.

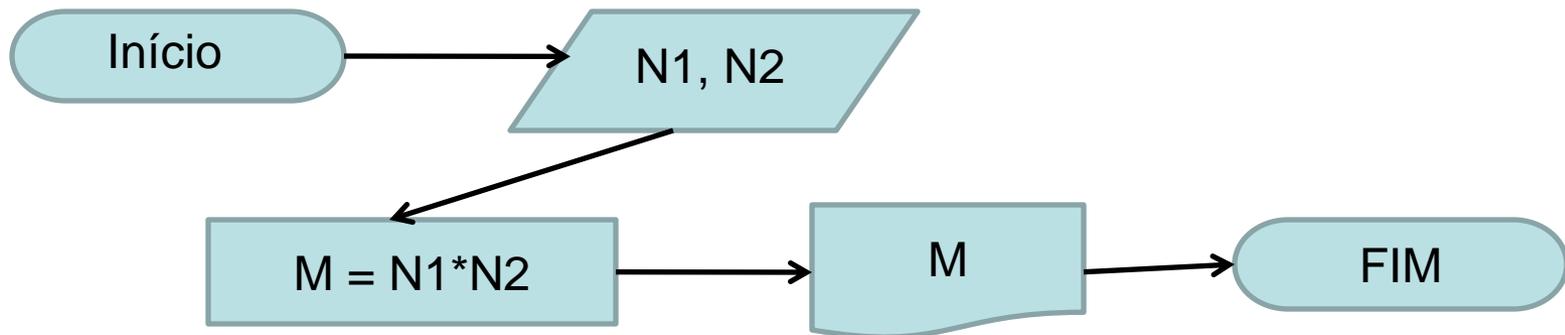
Algoritmo em descrição narrativa

PASSO 1 – Receber os dois números que serão multiplicados.

PASSO 2 – Multiplicar os números.

PASSO 3 – Mostrar o resultado obtido da multiplicação.

Algoritmo em fluxograma



Representação e documentação

Algoritmo em pseudocódigo

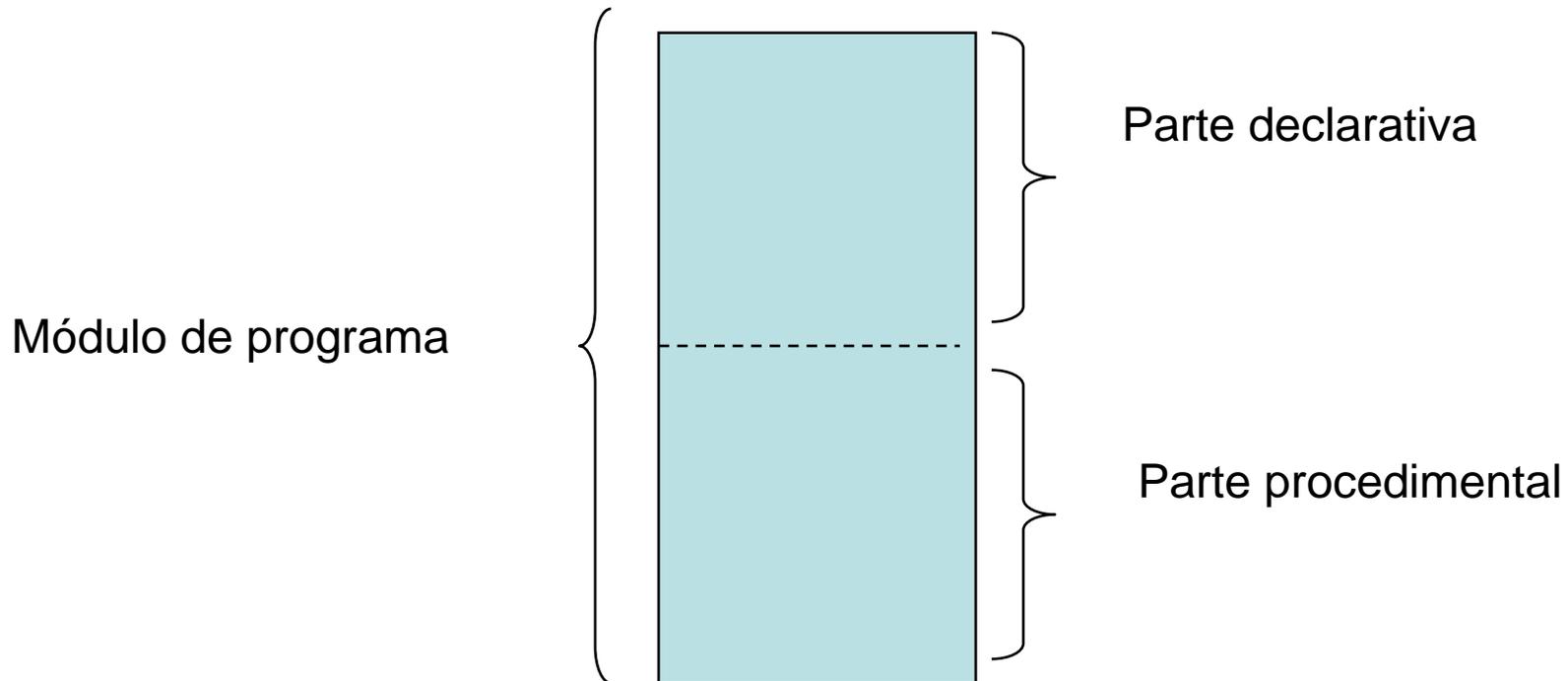
```
ALGORITMO  
DECLARE N1, N2, M NUMÉRICO  
ESCREVA "Digite dois números"  
LEIA N1, N2  
M ← N1*N2  
ESCREVA "Multiplicação = ", M  
FIM_ALGORITMO
```

Representação e documentação

- ❖ Os comandos nas linguagens de programação pertencem a três categorias.
 - **Comandos declarativos:** definem uma terminologia personalizada a ser utilizada no programa. Por exemplo, nomes associados aos dados.
 - **Comandos imperativos:** descrevem os passos dos algoritmos subjacentes.
 - **Comentários:** facilitam a leitura de um programa.
-

Representação e documentação

- ❖ **Parte declarativa de um programa:** são usados comandos declarativos para descrever a terminologia.
- ❖ **Parte processual (procedimental):** são utilizados comandos imperativos para descrever ações a serem executadas.



Representação e documentação

❖ Variáveis

- São identificadores de posições de memória cujo valor armazenado está sujeito a alterações.
- Possui nome (ou identificador) e tipo (numéricos, lógicos, literais ou caracteres).

❖ Identificadores

- Nomes das variáveis, constantes, rotinas, programas, etc.
 - Podem ser formados utilizando caractere sublinhado, números e letras (maiúsculas ou minúsculas).
 - Uma letra ou o caractere sublinhado podem ser o primeiro caractere em um identificado.
 - Espaços em branco e caracteres especiais (@,\$,+, -, %, !) não podem ser utilizados em um identificador.
 - Também não podem ser utilizadas palavras reservadas.
-

Representação e documentação

❖ Tipo de dados

- ❑ Podem ser numéricos, literal ou lógico.
 - ❑ Um tipo define determinado dado, permitindo saber quais operações podem ser executadas sobre determinado tipo de dado.
 - ❑ Por exemplo, é possível somar dois valores numéricos, mas não é possível somar um número e uma frase (cadeia de caracteres).
 - ❑ O tipo de dado adequado deve ser manipulado pelo algoritmo.
 - ❑ Por exemplo, não podemos utilizar um tipo inteiro para tratar valores reais.
-

Representação e documentação

❖ Tipos de Dados Numéricos

❑ Podem ser inteiros ou reais.

❑ Exemplo:

Inteiros	Reais
-23	23.45
98	346.89
0	0.0
237	-34.88
-2	-247.0

Representação e documentação

❖ Tipos de Dados Lógicos

- ❑ Também chamados booleanos.
- ❑ Assumem valores verdadeiro ou falso.

❖ Tipos de Dados Literais ou Caracteres

- ❑ Formados por um único caractere ou por uma cadeia de caracteres.
- ❑ Os caracteres podem ser letras maiúsculas, minúsculas, números e caracteres especiais (&, #, @, ?, +).
- ❑ Os números, enquanto caracteres, não podem ser usados para cálculos.
- ❑ Exemplo: 'aluno', '123', '@ internet', '0.34', '1+2'.

Representação e documentação

❖ Comentários

- Formas sintáticas que permitem a inserção de textos explicativos no programa.
 - A documentação resultante é chamada documentação interna que é ignorada pelo tradutor, ou seja, não afeta o programa do ponto de vista da máquina.
 - Sem essa documentação, programas grandes e complexos podem facilmente ultrapassar os limites da capacidade humana de compreensão.
 - São utilizados para descrever um algoritmo ou parte dele;
 - São importantes para aumentar a legibilidade ou entendimento da solução expressa por um algoritmo;
 - Podem ser usados para indicar o significado das variáveis e constantes utilizadas ;
-

Representação e documentação

❖ Estrutura Condicional Simples

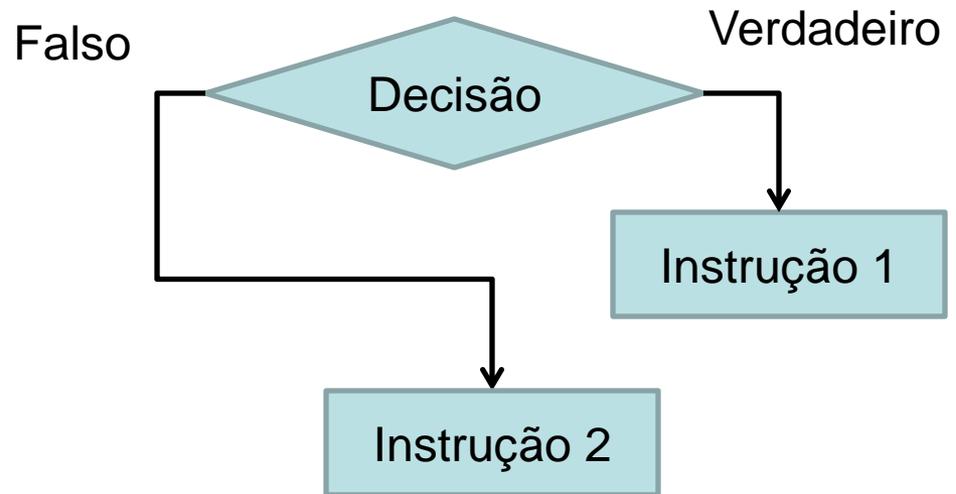
Algoritmo em pseudocódigo

SE <Decisão>

ENTÃO Instrução 1

Instrução2

Algoritmo em fluxograma



Representação e documentação

❖ Estrutura Condicional Composta

Algoritmo em pseudocódigo

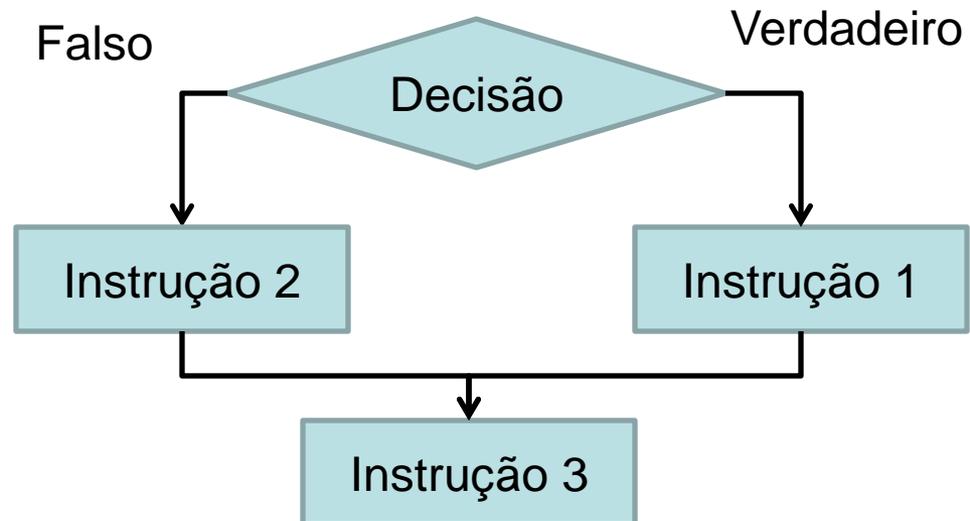
SE <Decisão>

ENTÃO Instrução 1

SENÃO Instrução2

Instrução 3

Algoritmo em fluxograma



Exercício

Apresente os algoritmos em descrição narrativa, fluxograma e pseudocódigo.

1. Faça um algoritmo que exiba o resultado da divisão de dois números.
 2. Faça um algoritmo para calcular a média aritmética entre duas notas de um aluno e mostrar sua situação, que pode ser aprovado ou reprovado.
 3. Faça um algoritmo para calcular o novo salário de um funcionário. Sabe-se que os funcionários que recebem atualmente salário de até R\$500 terão aumento de 20%; os demais terão aumento de 10%.
-