

Pseudocódigo – Exercício 6

Elabore um algoritmo que receba como entrada o valor do saque realizado pelo cliente de um banco e retorne quantas notas de cada valor serão necessárias para atender ao saque com a menor quantidade de notas possível. Serão utilizadas notas de 100, 50, 20, 10, 5, 2 e 1 reais.

algoritmo "exercício 6"

var

saque: inteiro

inicio

escreva ("Entre com o valor do saque: ")

leia (saque)

escreval ("Número de cédulas de R\$ 100: ",saque\100)

saque <- saque - saque\100*100

escreval ("Número de cédulas de R\$ 50: ",saque\50)

saque <- saque%50

escreval ("Número de cédulas de R\$ 20: ",saque\20)

saque <- saque%20

escreval ("Número de cédulas de R\$ 10: ",saque\10)

saque <- saque%10

escreval ("Número de cédulas de R\$ 5: ",saque\5)

saque <- saque%5

escreval ("Número de cédulas de R\$ 2: ",saque\2)

saque <- saque%2

escreval ("Número de cédulas de R\$ 1: ",saque)

74 fimalgoritmo



Pseudocódigo – Exercício 7

Construa um algoritmo para ler um número inteiro, positivo de três dígitos, e gerar outro número formado pelos dígitos invertidos do número lido.

Ex: NumeroLido = 123
 NumeroGerado = 321

Dica: Observe os resultados das funções Quociente e Resto de um número por 10.

algoritmo "exercício 7.1"

var

numero: inteiro

inicio

**escreva ("Entre com um número inteiro positivo
com três dígitos: ")**

leia (numero)

**escreval ("Número resultante da inversão dos
dígitos: ", (numero%10*100)+
(numero\10%10*10)+(numero\10\10))**

fimalgoritmo

algoritmo "exercício 7.2"

var

numero: inteiro

inicio

**escreva ("Entre com um número inteiro positivo
com três dígitos: ")**

leia (numero)

**escreval ("Número resultante da inversão dos
dígitos: ", (numero%10*100)+(numero%100-
numero%10)+(numero\100))**

fimalgoritmo

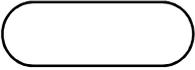
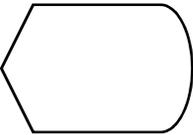
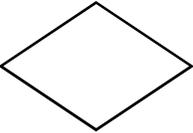
Fluxograma

Analisaremos agora o método de representação de algoritmos denominado fluxograma.

Conceitualmente um fluxograma é um tipo de diagrama, e pode ser entendido como uma representação esquemática de um processo, constitui uma representação gráfica que ilustra de forma descomplicada a sequência de execução dos elementos que o compõem. Podemos entendê-lo, na prática, como a documentação dos passos necessários para a execução de um processo qualquer.

Veremos agora alguns símbolos empregados na construção de fluxogramas.

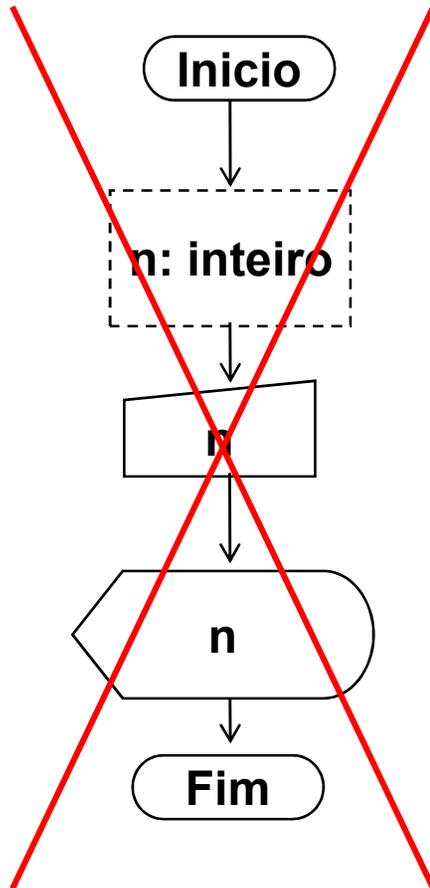
Fluxograma

Símbolo	Nome	Descrição
	Terminador	Indica o início e o fim do fluxo do algoritmo.
	Seta de fluxo	Indica o sentido do fluxo de execução do algoritmo. É através dela que os símbolos do fluxograma são conectados.
	Declaração	Delimita a seção de declaração de variáveis.
	Entrada de dados	Corresponde à instrução de entrada de dados através do teclado.
	Atribuição	Símbolo utilizado para indicar cálculos e atribuição de valores.
	Saída de dados	Corresponde à instrução de saída de dados. Os dados serão exibidos na tela do computador.
	Desvio condicional	Divide o fluxo do programa em dois caminhos, dependendo do teste lógico que fica dentro do losango.

Conjunto de símbolos utilizados em fluxogramas

Exemplo de Fluxograma

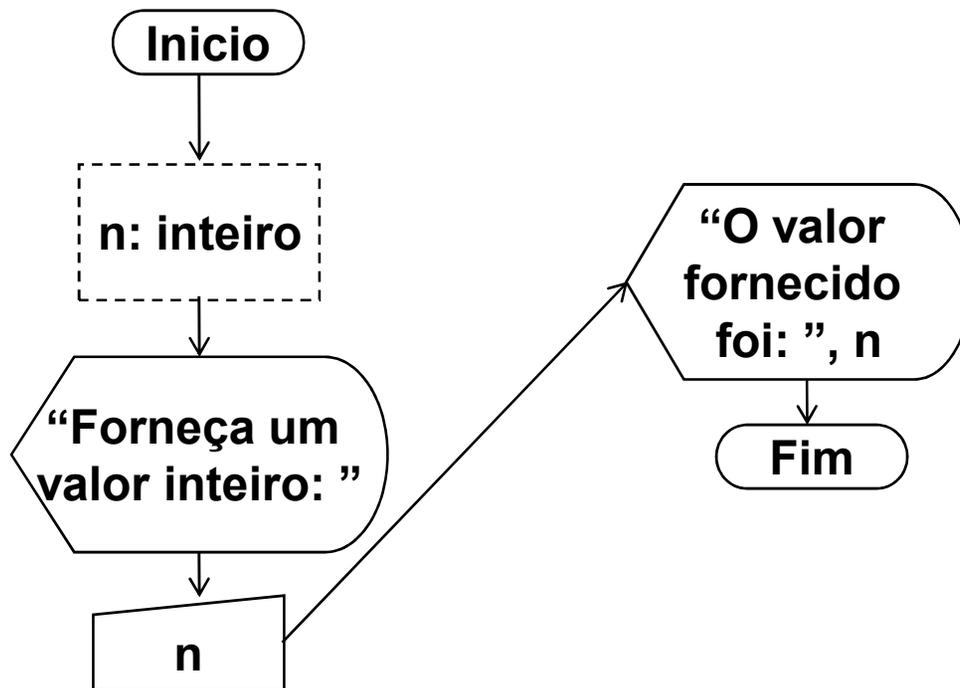
De forma similar à análise feita com pseudocódigo, iniciaremos nossa análise por um fluxograma que efetua a leitura, através do teclado, de um valor inteiro e o retorna no monitor.



Símbolo	Nome	Descrição
	Terminador	Indica o início e o fim do fluxo do algoritmo.
	Seta de fluxo	Indica o sentido do fluxo de execução do algoritmo. É através dela que os símbolos do fluxograma são conectados.
	Declaração	Delimita a seção de declaração de variáveis.
	Entrada de dados	Corresponde à instrução de entrada de dados através do teclado.
	Atribuição	Símbolo utilizado para indicar cálculos e atribuição de valores.
	Saída de dados	Corresponde à instrução de saída de dados. Os dados serão exibidos na tela do computador.
	Desvio condicional	Divide o fluxo do programa em dois caminhos, dependendo do teste lógico que fica dentro do losango.

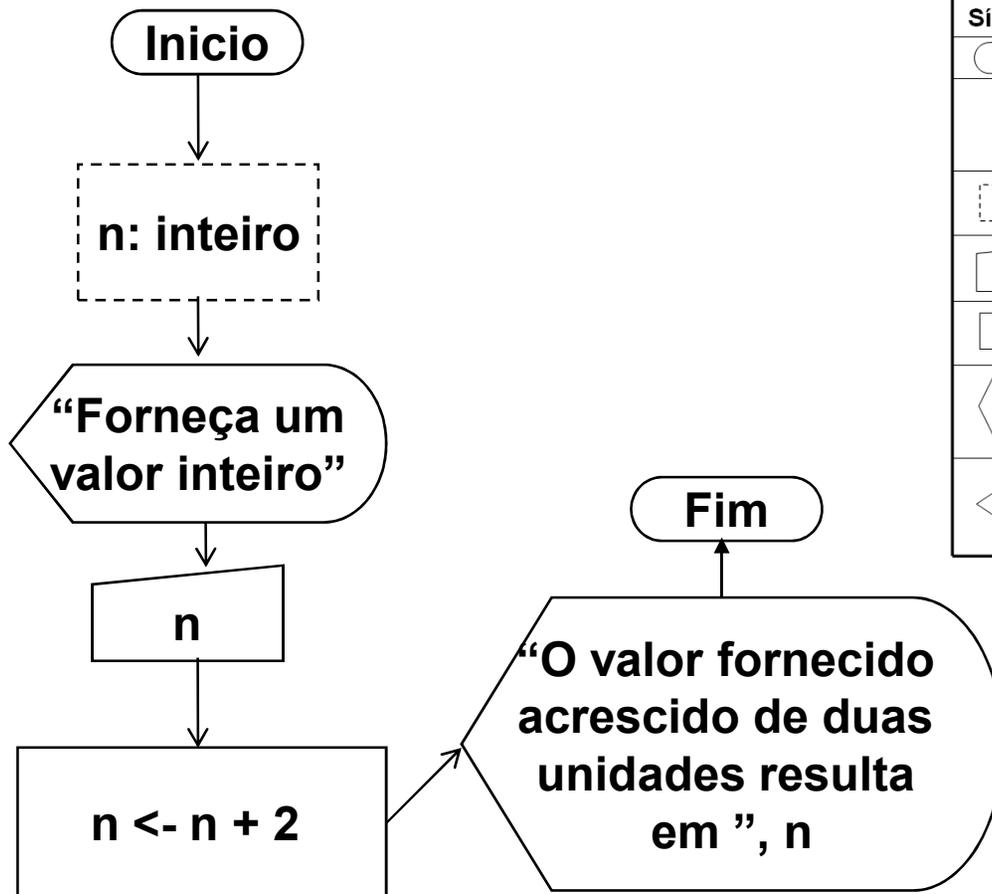
Exemplo de Fluxograma

De forma similar à análise feita com pseudocódigo, iniciaremos nossa análise por um fluxograma que efetua a leitura, através do teclado, de um valor inteiro e o retorna no monitor.



Exemplo de Fluxograma

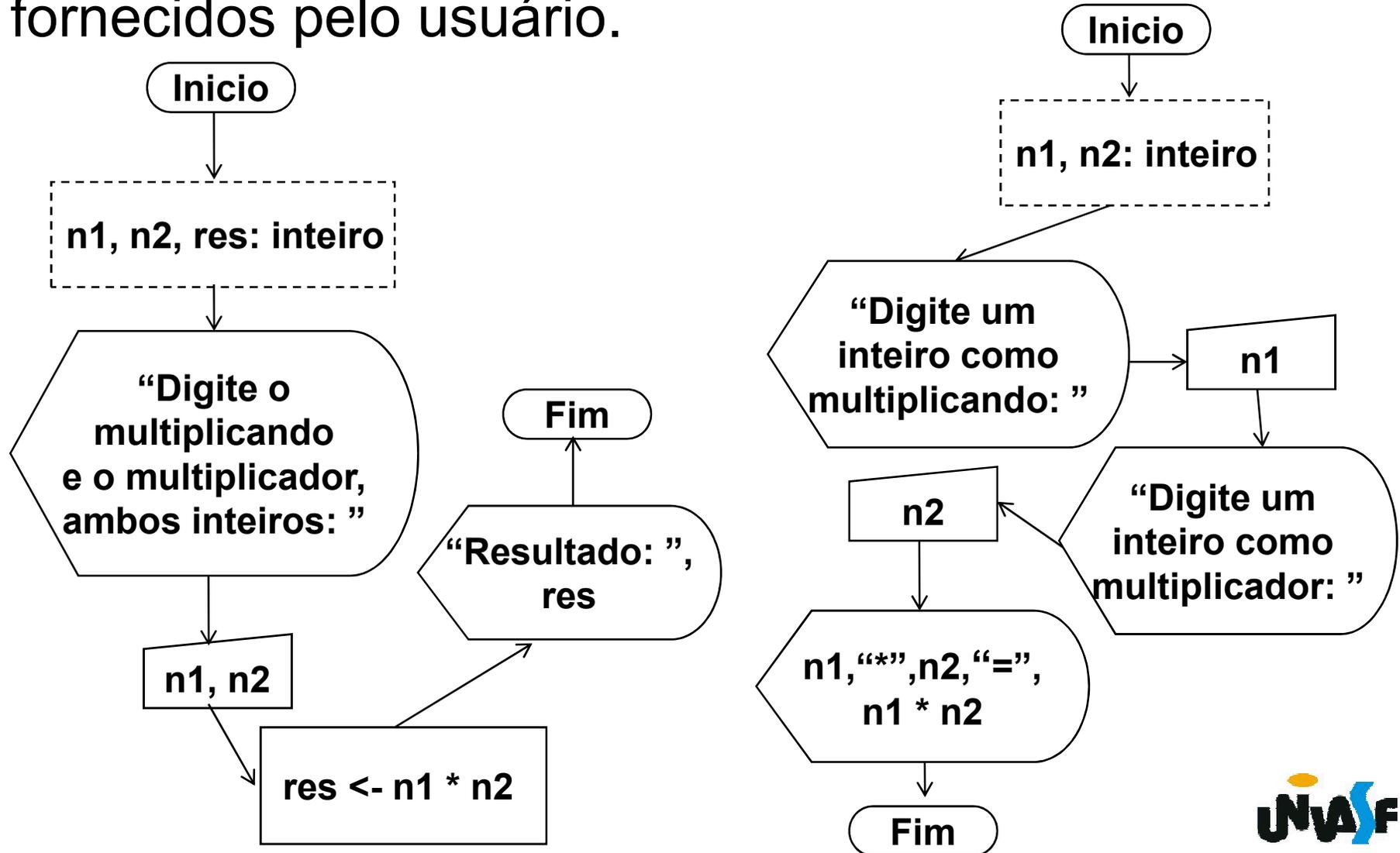
Observaremos agora um fluxograma que recebe um valor inteiro, através da entrada padrão, e acresce duas unidades a este exibindo o resultado na saída padrão.



Símbolo	Nome	Descrição
	Terminador	Indica o início e o fim do fluxo do algoritmo.
	Seta de fluxo	Indica o sentido do fluxo de execução do algoritmo. É através dela que os símbolos do fluxograma são conectados.
	Declaração	Delimita a seção de declaração de variáveis.
	Entrada de dados	Corresponde à instrução de entrada de dados através do teclado.
	Atribuição	Símbolo utilizado para indicar cálculos e atribuição de valores.
	Saída de dados	Corresponde à instrução de saída de dados. Os dados serão exibidos na tela do computador.
	Desvio condicional	Divide o fluxo do programa em dois caminhos, dependendo do teste lógico que fica dentro do losango.

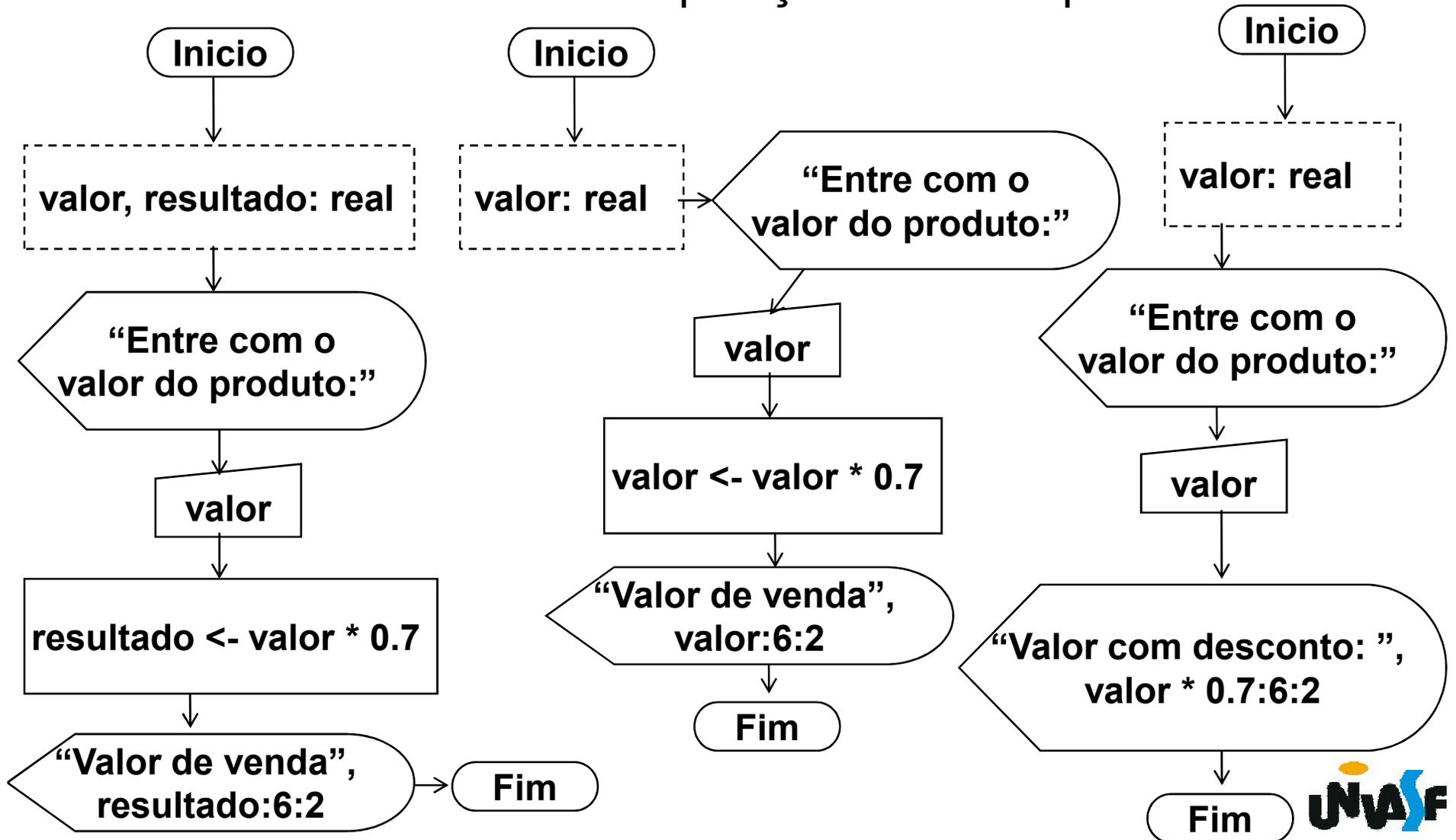
Exercício de Fluxograma

Construa um fluxograma para obter o resultado da multiplicação de dois números inteiros quaisquer fornecidos pelo usuário.



Exercício de Fluxograma

Gere um fluxograma que aplique um desconto de 30% sobre o valor de um produto, recebido como entrada, e retorne o resultado da manipulação na saída padrão.



Estruturas de Controle de Fluxo

Os algoritmos desenvolvidos até o momento constituem uma sequência de ações que sempre são executadas em sua totalidade indiferente de qual(is) seja(m) o(s) valor(es) da(s) entrada(s).

Contudo, para a resolução de determinados problemas ou para a execução de determinadas tarefas é necessária a realização de um conjunto distinto de ações e este conjunto é definido com base em uma análise da(s) entrada(s).

Um exemplo cotidiano de uma destas situações é um algoritmo capaz de efetuar o cálculo do imposto de renda devido por um determinado contribuinte. Neste caso, dependendo da quantidade de dependentes, do valor de sua renda e outras fatores o cálculo será feito de formas distintas.

Estruturas de Controle de Fluxo

Em função do que foi mencionado foram criadas as estruturas de controle de fluxo, as quais são fundamentais para a construção de algoritmos complexos. Estas permitem que o programador especifique a sequência de instruções que será executada.

1. Instrução condicional simples

Sintaxe: ...

se (*<expressão-lógica>*) então
<sequência-de-comandos>

fimse

...

Estruturas de Controle de Fluxo

Pseudocódigo/Exercício – Construa o pseudocódigo de um algoritmo para obter o resultado da divisão de dois números inteiros quaisquer.