



Programação de Computadores I

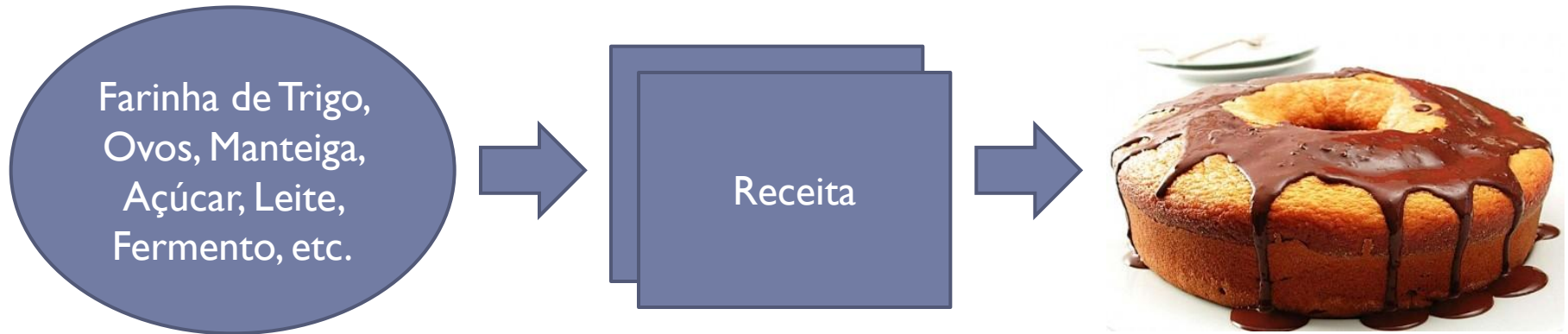
Fluxogramas

PROFESSORA CINTIA CAETANO

Problemas & Algoritmos

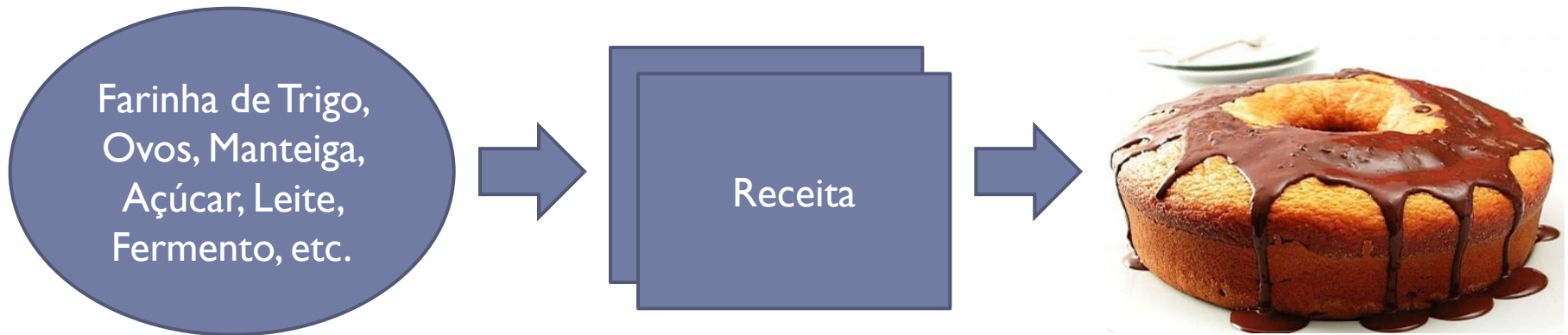
- ▶ Para resolver um problema através dum computador é necessário encontrar em primeiro lugar uma maneira de descrevê-lo de uma forma clara e precisa.
- ▶ É também preciso que encontremos uma seqüência de passos que conduzam à sua resolução. Esta seqüência de passos é designada por algoritmo.
- ▶ • A noção de algoritmo é central para toda a informática.
- ▶ • A criação de algoritmos para resolver os problemas é uma das maiores dificuldades, mas também um dos desafios mais atrativos, dos iniciados em programação em computadores.

Problema: Fazer um bolo?



- ▶ Uma receita é uma descrição dum conjunto de passos ou ações que fazem a combinação dum conjunto de ingredientes com vista a obter um produto gastronômico particular.

Algoritmo: Como fazer um bolo?



► Algoritmo (receita de bolo):

1. Bater duas claras em castelo;
2. Adicionar duas gemas;
3. Adicionar um xícara de açúcar;
4. Adicionar duas colheres de manteiga;
5. Adicionar uma xícara de leite de coco;
6. Adicionar farinha e fermento;
7. Colocar numa forma e levar ao forno em lume brando.

Um algoritmo opera sobre um conjunto de entradas (farinha ovos, fermento, etc. no caso do bolo) de modo a gerar uma saída que seja útil (ou agradável) para o utilizador (o bolo pronto).

Desenho de Algoritmos/Programas

- ▶ De um modo geral, considera-se que um algoritmo é uma descrição, passo-a-passo, de uma metodologia que conduz à resolução de um problema ou à execução de uma tarefa.
- ▶ A programação consiste na codificação precisa desse algoritmo, segundo uma linguagem de programação específica.
- ▶ Há, pois, que ter em consideração que existem três fases distintas na elaboração de programas:
 - ▶ A análise do problema (especificação do problema, análise de requisitos, pressupostos, etc.)
 - ▶ A concepção do algoritmo
 - ▶ A tradução desse algoritmo na linguagem de programação



Passos na construção de Algoritmos

- ▶ **Compreender o problema**
 - ▶ Identificar os dados de entrada
 - ▶ Identificar os dados de saída
- ▶ **Determinar o que é preciso para transformar dados de entrada em dados de saída:**
 - ▶ Usar a estratégia do dividir-para-conquistar
 - ▶ Observar regras e limitações
 - ▶ Identificar todas as ações a realizar
 - ▶ Eliminar ambigüidades
- ▶ **Construir o algoritmo**
- ▶ **Testar o algoritmo**
- ▶ **Executar o algoritmo**

Método Cartesiano de Dividir-Para-Conquistar

- ▶ Também é o conhecido por método descendente (top-down method) ou método de refinamento passo-a-passo.
- ▶ Este método consiste em dividir um problema em partes menores (ou sub-problemas) de modo a que seja mais fácil a sua resolução.
- ▶ Exemplo: Fazer sumo de laranja?
 - ▶ Lavar laranja;
 - ▶ Partir laranja ao meio;
 - ▶ Espremer laranja;
 - ▶ Filtrar o sumo;
 - ▶ Servir o sumo.
- ▶ Passo-a-passo, significa que cada passo é completado antes que o próximo comece.
- ▶ Exemplo: é impossível “ver telejornal” antes de executar por inteiro o passo anterior de “ligar a TV”

Características fundamentais de um Algoritmo

- ▶ Um algoritmo deve ter 5 características fundamentais:
 - ▶ **Finitude:** um algoritmo deve sempre terminar após um número finito de passos.
 - ▶ **Definição:** cada passo de um algoritmo deve ser precisamente definido. As ações devem ser definidas rigorosamente e sem ambigüidades.
 - ▶ **Entradas:** um algoritmo deve ter zero ou mais entradas, isto é quantidades que lhe são fornecidas antes do algoritmo iniciar.
 - ▶ **Saídas:** um algoritmo deve ter uma ou mais saídas, isto é quantidades que tem uma relação específica com as entradas.
 - ▶ **Eficiência:** um algoritmo deve ser eficiente. Isto significa que todas as operações devem ser suficientemente básicas de modo que possam ser em princípio executadas com precisão em um tempo finito por um ser humano usando papel e lápis.
- ▶ **NOTA:** Pode haver mais do que um algoritmo para resolver um problema. Por exemplo, para ir de casa até o trabalho, posso escolher diversos meios de transportes em função do preço, conforto, rapidez, etc.

Fluxogramas

- ▶ Um das formas de representar de algoritmos.
- ▶ Tem como objetivo de descrever o conjunto de passos necessários à resolução de um problema utilizando simbologia própria e linguagem comum.

Fluxogramas

- ▶ Os símbolos gerais mais utilizados são:



início e fim de fluxograma



entrada e saída de dados



conector na mesma página



conector para outra página



teste e decisão

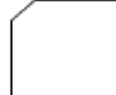
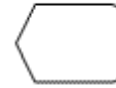


outras acções/sinistrações



inicialização
teste e actualização

Fluxogramas

			
Processo	Processo Alternativo	Decisão	Dados
			
Processo Pré-definido	Armazenamento Interno	Documento	Vários Documentos
			
Terminação	Preparação	Entrada Manual	Operação Manual
			
Conector	Conector Fora de Página	Cartão	Fita Perfurada
			
Somador	Ou	Agrupar	Classificar
			
Extrair	Mesclar	Dados Armazenados	Atraso
			
Armazenamento de Acesso Sequencial	Disco Magnético	Armazenamento de Acesso Direto	Exibir

Regras Básicas

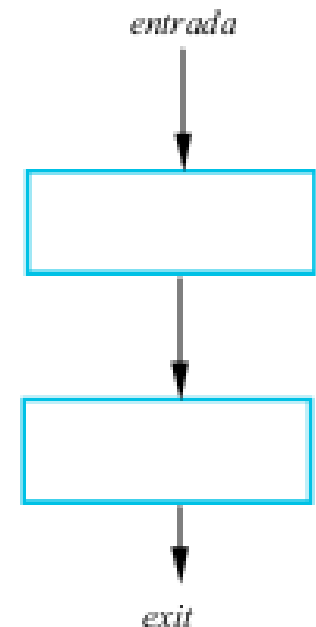
- ▶ Clareza e simplicidade na construção de um fluxograma são essenciais.
- ▶ Detalhes de sistema ou complexidades devem ser evitados logo no início do desenvolvimento do fluxograma.
- ▶ Texto explicativo de alguns trechos de comandos/instruções podem (e devem) ser incluídos, mas devem se breves.
- ▶ A combinação de fluxograma e textos explicativos costuma ser bastante eficiente na compreensão e solução de problemas.

Montagem de um fluxograma

- ▶ O fluxograma é a seqüência de um raciocínio lógico portanto, deve ter uma disposição dos diversos símbolos de forma lógica.
- ▶ Então, algumas regras simples devem ser seguidas:
 1. Iniciar do topo para a base da folha.
 2. Desenvolvimento lateral deve ser da esquerda para a direita, preferencialmente.
 3. Nas quebras de página, identificar numericamente a página de destino/retorno.
 4. Sempre empregar as flechas de sentido para maior clareza do fluxo de raciocínio.
 5. Numerar as páginas seqüencialmente.
 6. Escrever as operações dentro dos símbolos de forma clara e compacta.
 7. Fazer comentários quando a descrição das operações não for suficiente.

Sequência

- ▶ Numa seqüência é processado um conjunto de ações (ou instruções) em série.
- ▶ Não há qualquer possibilidade de alterar a ordem de processamento das ações.
- ▶ Ou seja, após processar a 1ª ação processa-se a 2ª, depois da 2ª processa-se a 3ª, e assim por diante até processar a última ação.



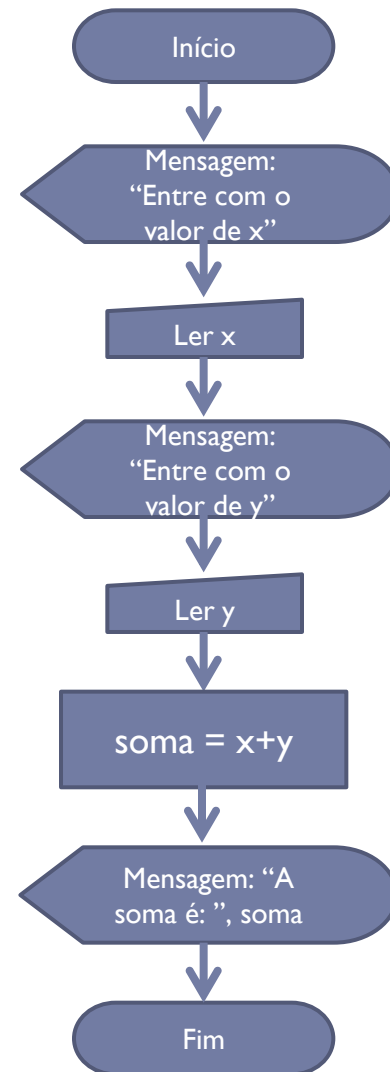
Exemplo 1

- ▶ Lê um valor e imprime na tela.



Exemplo 2

- Soma de dois números lidos a partir do teclado.

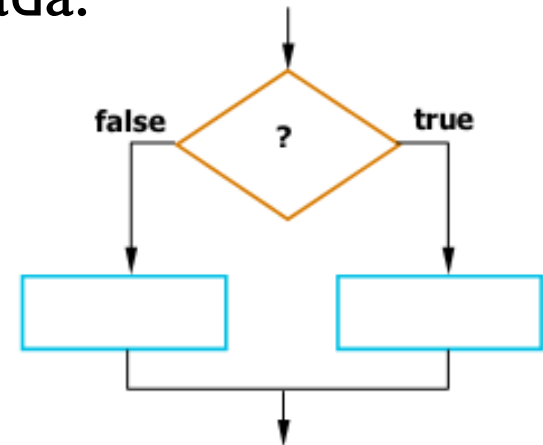


Exercícios

1. Calcule a média de dois valores e imprima na tela.
2. Calcule a área de um trapézio qualquer. Lembrando que:
$$\text{Área} = (\text{Base Maior} + \text{Base Menor}) \times \text{Atura} / 2.$$
3. Leia o nome do usuário e imprima a mensagem “Ola <nome do usuário>”. Ex. Ola João
4. Calcule o valor da conta de luz de uma pessoa. O valor do Kilowatt é R\$ 0,55.

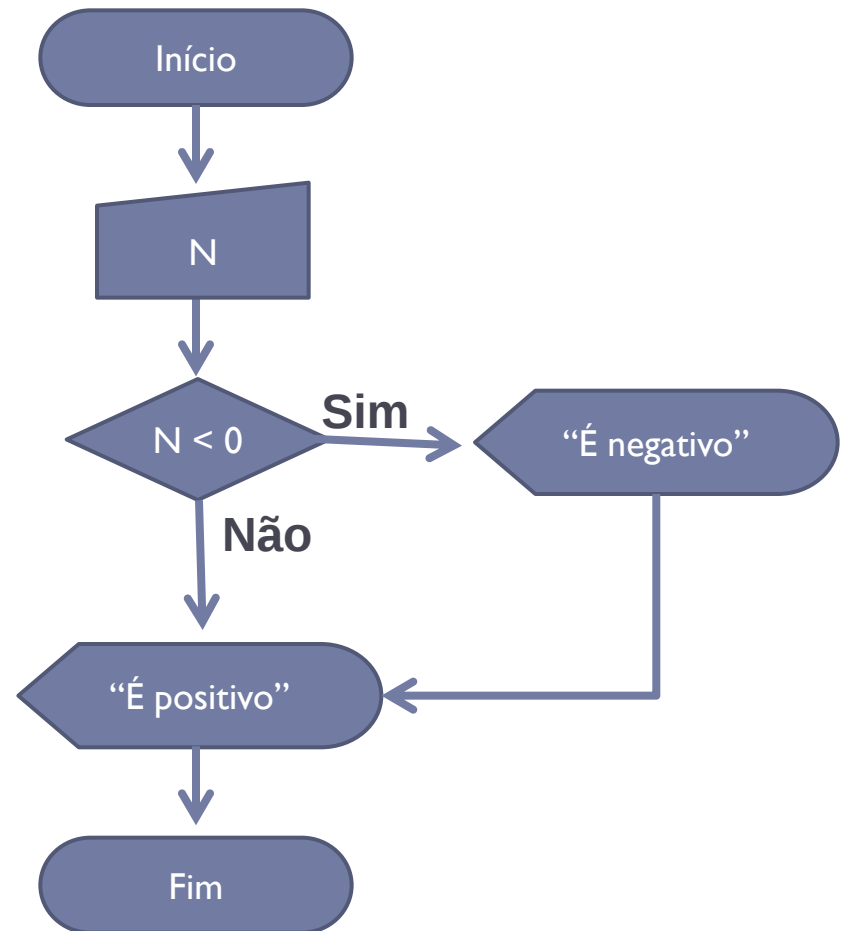
Seleção (com 2 vias)

- ▶ Uma estrutura de seleção é também designada por estrutura de decisão.
- ▶ Neste caso, o fluxo de processamento segue por 1 das 2 vias, dependendo do valor lógico (verdadeiro ou falso) da expressão avaliada no início da estrutura.
- ▶ Se o fluxo de processamento só passa por 1 via, então só uma das ações é realizada ou processada.



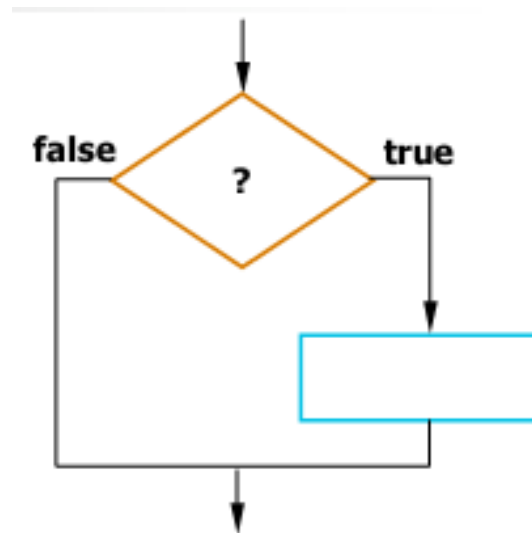
Exemplo 3

- ▶ Verificar se o número é menor que zero.
- ▶ Se for verdade, imprime a mensagem “É negativo”.
- ▶ Caso contrário, imprime a mensagem “É positivo”.



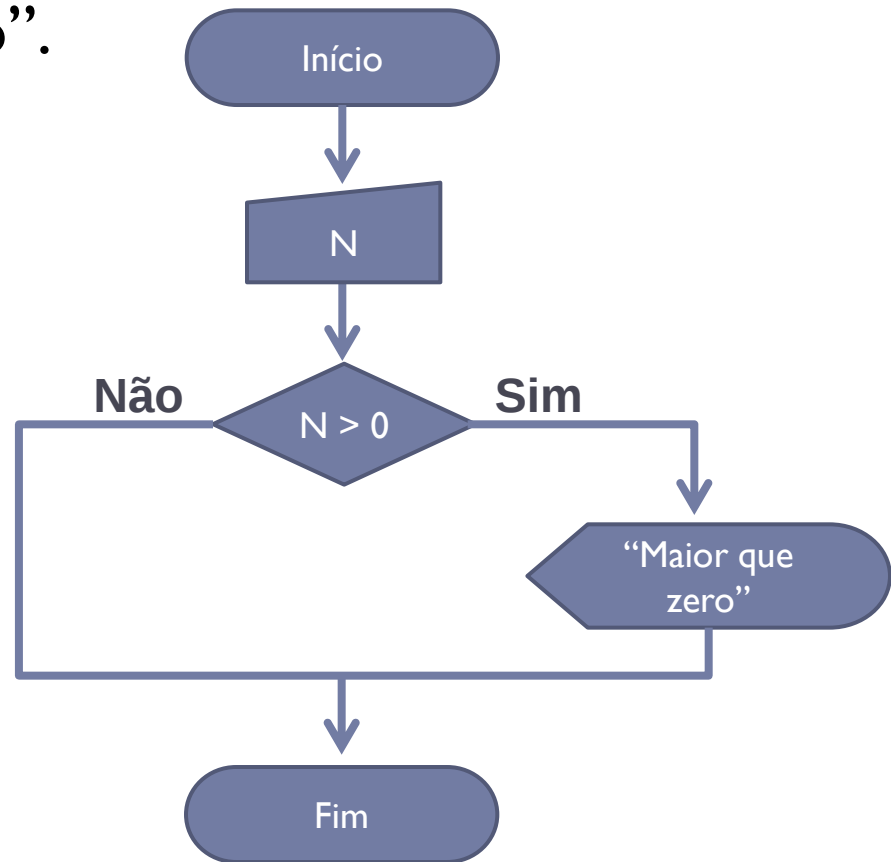
Seleção (com 1 via)

- ▶ Neste caso, se a expressão lógica tiver resultado false (falso), nenhuma ação é processada dentro da estrutura de seleção.
- ▶ Só é processada uma ação dentro da estrutura de seleção se a expressão lógica for true (verdadeiro).



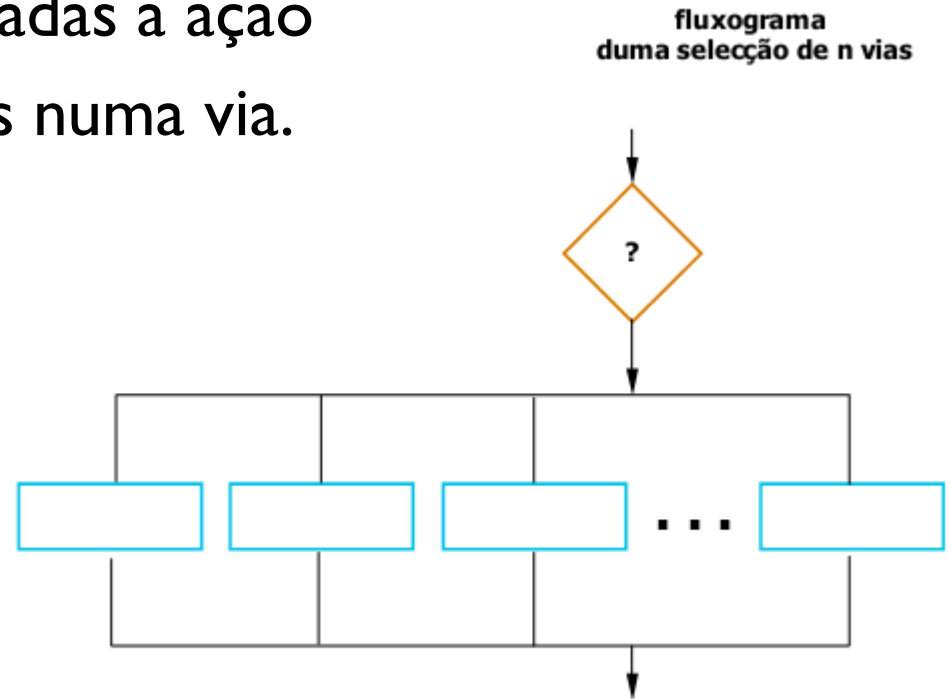
Exemplo 4

- Se o valor recebido for maior que zero ele informa na tela que o valor “é aceito”.



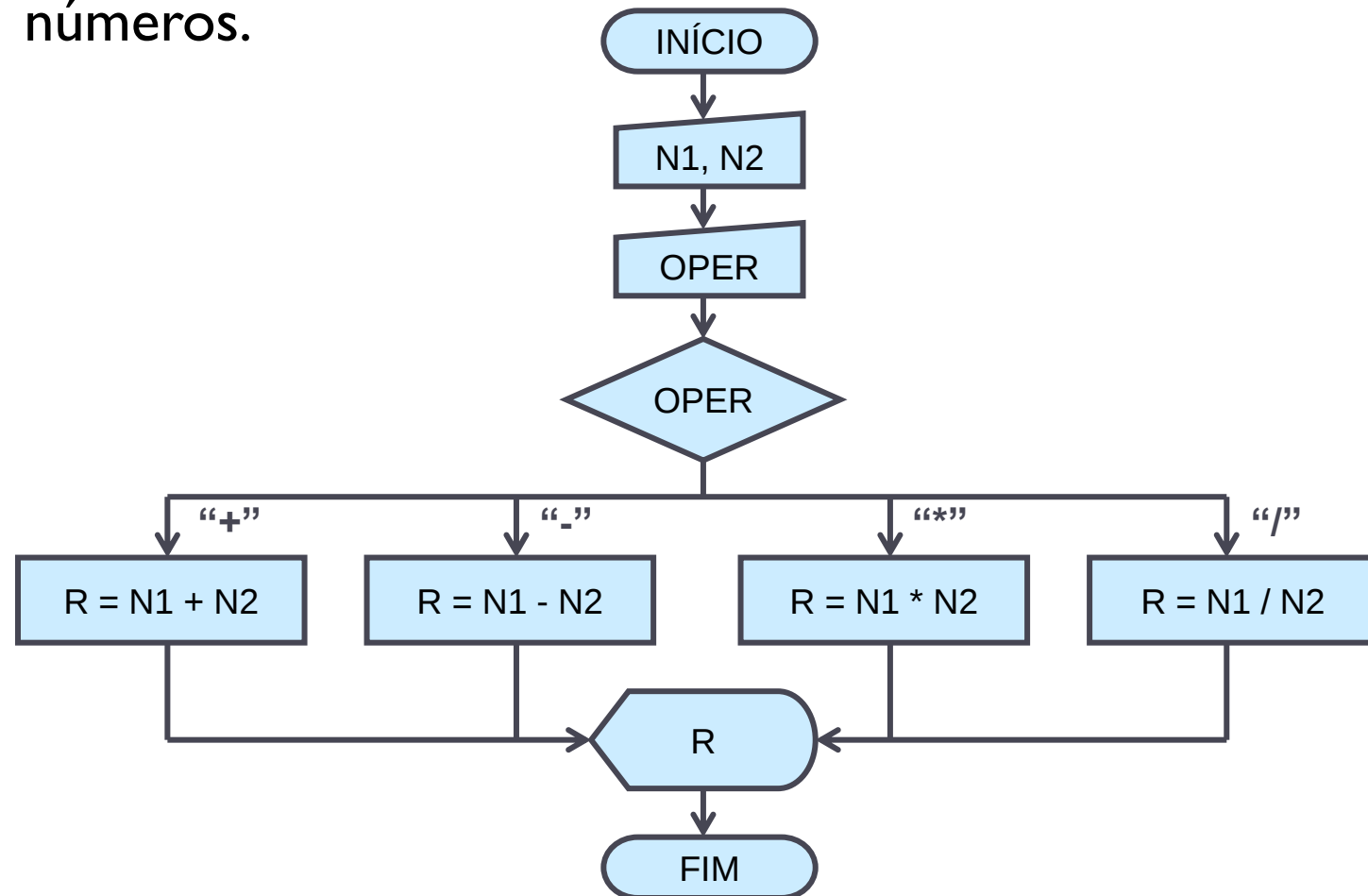
Seleção (com n-vias)

- ▶ Neste caso, a decisão não é feita com base numa expressão lógica porque há mais do que 2 resultados possíveis.
- ▶ Também só são processadas a ação ou as ações encontradas numa via.



Exemplo 5

- Escolher uma das 4 operações básicas para aplicar em dois números.



Exercícios

- ▶ Elabore um fluxograma que verifique se um número inteiro é par ou ímpar.
- ▶ Elabore um fluxograma que leia dois números e verifique o maior deles.
- ▶ Receba duas notas de um aluno e verifique se o aluno está aprovado ($\geq 6,0$) ou reprovado ($< 6,0$). Imprima na tela a mensagem.
- ▶ Ler uma nota e verificar se a mesma esta no intervalo $[9,10[$, caso positivo ela deve ser retificada para 9.5.
- ▶ Ler o nome de uma pessoa, se o nome da pessoa for igual a João, imprima a mensagem “Usuário encontrado!”.

Exercícios

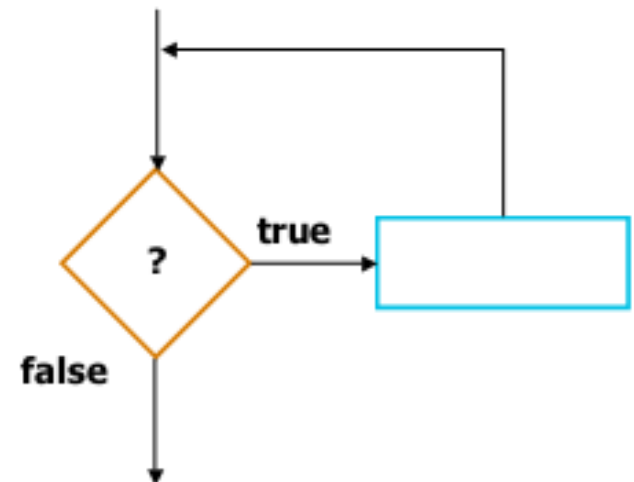
- ▶ Em uma loja de CD's existem apenas quatro tipos de preços que estão associados a cores. Assim os CD's que ficam na loja não são marcados por preços e sim por cores. Desenvolva o algoritmo que a partir da entrada da cor o software mostre o preço. A loja está atualmente com a seguinte tabela de preços.
 - ▶ Verde 10,00
 - ▶ Azul 20,00
 - ▶ Amarelo 30,00
 - ▶ Vermelho 40,00

Exercícios

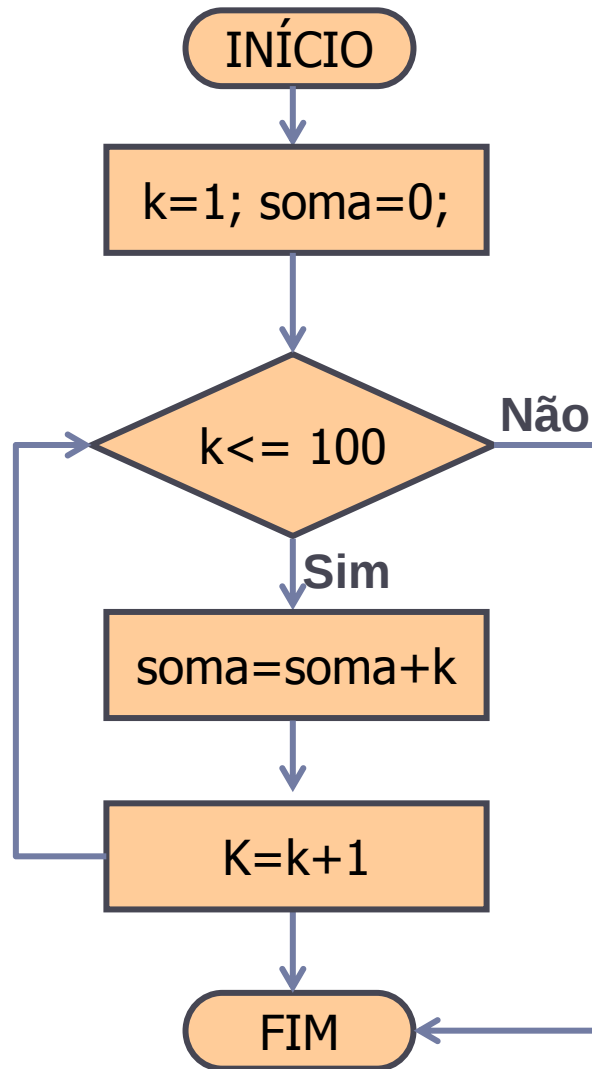
- ▶ Em uma escola, a média final é dada pela média aritmética de três notas. E a mesma tem o seguinte esquema de avaliação:
 - ▶ 0 – 4.9 Aluno em recuperação
 - ▶ 5 – 6.9 Aluno em prova final
 - ▶ 7 – 10 Aluno passado por média
- ▶ Desenvolva um algoritmo que a partir da entrada das três notas mostre a situação do aluno. No caso do aluno em recuperação e prova final, mostre também quanto o aluno irá precisar para passar. No caso da recuperação a nota necessária para passar é dada por $10 - \text{Média} + 2$ e na prova final é dado por $10 - \text{Média}$.

Repetição com Teste à Cabeça

- ▶ Neste caso, também há a necessidade de tomar uma decisão com base no valor lógico duma expressão.
- ▶ No entanto, a mesma ação será executada repetidamente enquanto o resultado da expressão lógica se mantiver verdadeiro (true).
- ▶ O teste (da expressão lógica) precede a ação. Diz-se, por isso, que o teste é à cabeça.
- ▶ O teste é importante porque funciona como uma condição de paragem (a false) dos ciclos ou repetições.

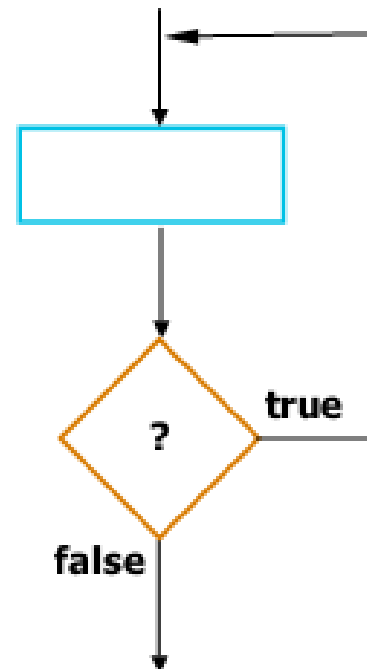


Exemplo 6

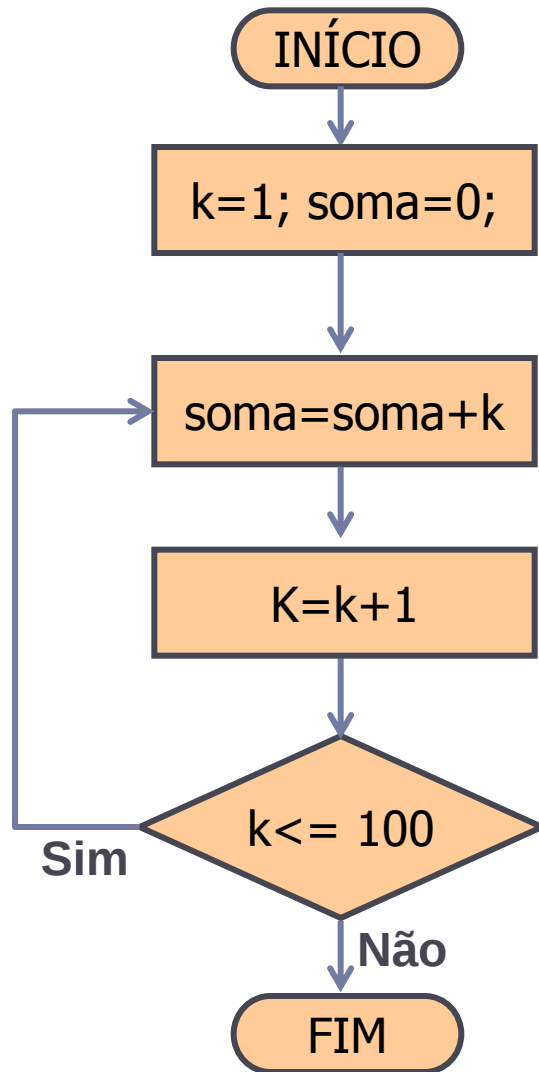


Repetição com Teste à Cauda

- ▶ Esta estrutura de repetição é em tudo idêntica à anterior. A diferença é que o teste é feito após o processamento da ação.
- ▶ O teste (da expressão lógica) sucede a ação. Diz-se, por isso, que o teste é à cauda.

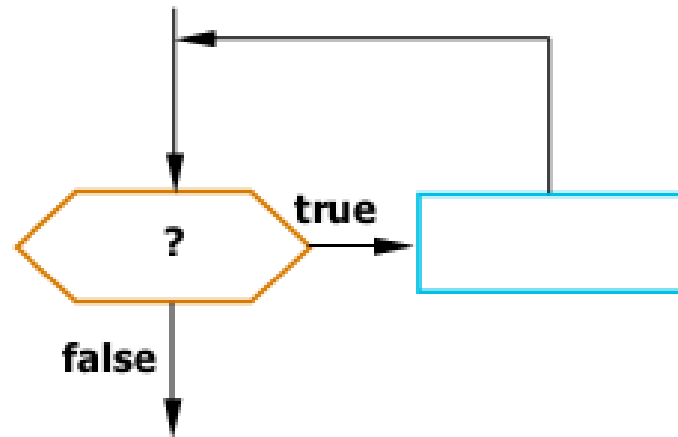


Exemplo 7

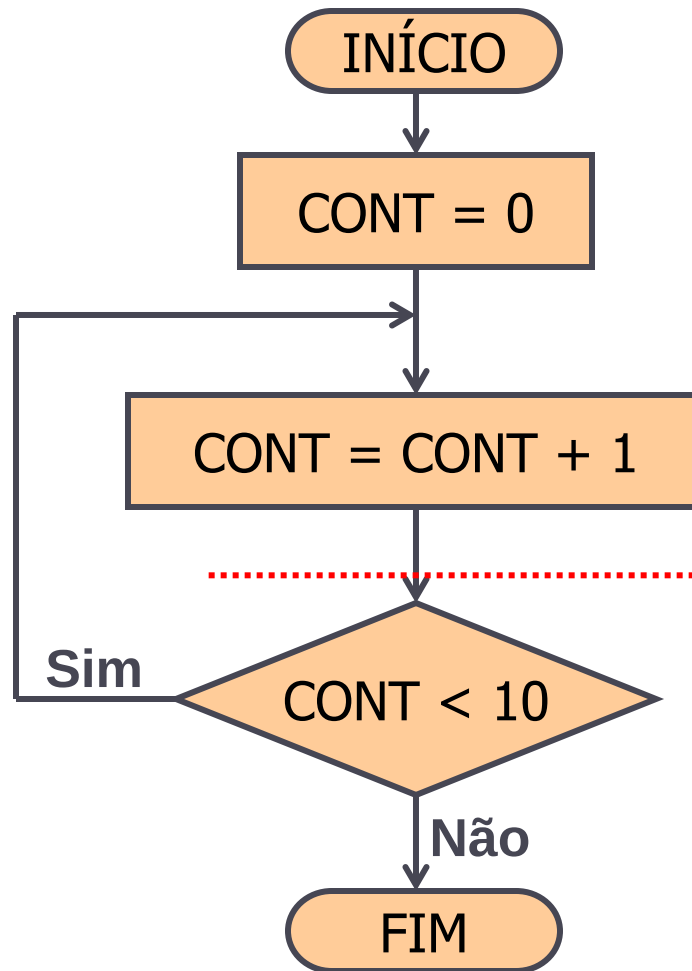


Repetição com Número Pré-definido de ciclos

- ▶ Esta estrutura de repetição é em tudo idêntica às anteriores.
- ▶ O teste é feito à cabeça.
- ▶ A diferença é que é logo à partida especificado o número de ciclos (ou iterações) que serão efetuados, ou seja, o número de vezes que a ação será processada.



Exemplo 8



Neste ponto do algoritmo podemos incluir qualquer conjunto de instruções que quisermos repetir 10 vezes.

Exercícios

1. Faça a leitura de dez números e calcule a média dos números positivos.
2. Monte um fluxograma para representar o preparo de um ovo frito.

Exercícios

1. Desenvolva um algoritmo capaz de encontrar o menor dentre 3 números inteiros quaisquer dados pelo teclado.
2. Desenvolva um algoritmo que a partir da entrada das três notas mostre a situação do aluno. No caso do aluno em recuperação e prova final, mostre também quanto o aluno irá precisar para passar. No caso da recuperação a nota necessária para passar é dada por $10 - \text{Média} + 2$ e na prova final é dado por $10 - \text{Média}$.

Exercícios

1. Elabore um fluxograma que calcule e exiba a média de 500 números digitados pelo usuário.
2. Elabore um fluxograma que calcule e exiba a soma dos números contidos entre zero e o número digitado.
3. Elabore um fluxograma que calcule e exiba a soma dos números pares contidos entre zero e o número digitado.
4. Elabore um fluxograma que permita a entrada de dez valores e calcule o produto de todos eles.