

[< VOLTAR](#)

Formas de representação de algoritmos

Apresentar as formas de representação existentes para os algoritmos.

NESTE TÓPICO

- > Introdução
- > Descrição Narrativa
- > Diagrama de Blocos (Fluxograma)
- >

Marcar
tópico



Introdução

A tarefa de especificar os algoritmos para representar um programa consiste em detalhar os dados que serão processados pelo programa e as instruções que vão operar sobre esses dados. Por isso, é importante formalizar a descrição dos algoritmos segundo alguma convenção para que todas as pessoas envolvidas na sua criação possam entendê-lo da mesma forma.

Dentre as formas de representação de algoritmos podemos citar:

- Descrição Narrativa.
- Diagrama de Blocos (Fluxograma).
- Português Estruturado (pseudocódigo).

Descrição Narrativa

Nada mais é do que descrever, utilizando uma linguagem natural (língua portuguesa) os algoritmos, as ações a serem realizadas no tratamento dos dados de entrada para os resultados de saída na resolução do problema proposto.

As vantagens da descrição narrativa são:



- não tem esforço do aprendizado.
- o português é bastante conhecido por nós.

As desvantagens da descrição narrativa são:

- interpretação diferente (ambiguidade ou dupla interpretação).
- imprecisão.
- pouca confiabilidade (a imprecisão acarreta a desconfiança).
- extensão (normalmente, escreve-se muito para dizer pouca coisa).

No exemplo a seguir, podemos observar o algoritmo para a soma de dois números inteiros representado em descrição narrativa.

1. Informe o primeiro número
2. Informe o segundo número
3. Some os dois números
4. Mostre o resultado da soma.

Como o algoritmo representado em descrição narrativa pode dar margem a más interpretações (ambiguidades), esta forma não é muito utilizada.

Diagrama de Blocos (Fluxograma)

Diagrama de blocos (fluxograma) é uma ferramenta usada e desenvolvida pelos profissionais que estão envolvidos diretamente com a programação, tendo como objetivo descrever a sequência de operações a serem efetuadas em um processamento computacional.

Um diagrama de blocos é, basicamente, a especificação de um algoritmo (um processo), enquanto que um fluxograma serve para descrever qualquer coisa, não necessariamente um fluxo de um algoritmo, seria uma definição mais genérica.

Após a elaboração do diagrama de blocos é realizada a codificação do programa.

O Diagrama de bloco é uma forma padronizada e eficaz para representar algoritmos usando uma representação com símbolos gráficos preestabelecidos. Sua principal função é a de facilitar a visualização dos passos de um processamento.

As vantagens do diagrama de blocos são:









- Entendimento, pois qualquer representação gráfica sempre é mais bem entendida do que a representação por escrito.
- Padrão mundial.

As desvantagens do diagrama de blocos são:



- necessidade do aprendizado de todos os símbolos.
- interpretação diferente (imprecisão).
- pouca confiabilidade (a imprecisão acarreta a desconfiança).
- extensão (normalmente, escreve-se muito para dizer pouca coisa).
- complica-se à medida que o algoritmo cresce.

Existem diversos símbolos em um diagrama de blocos. Na Tabela a seguir podemos observar alguns dos símbolos que iremos utilizar:

	Terminação - Símbolo que identifica o início ou o fim do fluxo do processamento
	Seta de Fluxo de Processamento - Serve para conexão de símbolos
	Processamento - Processamento em geral, ou seja, indicar cálculos e efetuar atribuições de valores, ou qualquer manipulação de dados que representada por um bloco único.
	Dados (Entrada ou saída de dados) - Um dispositivo qualquer de entrada ou saída de dados, como fornecedor de informações para processamento, gravação e outros.
	Decisão - Indica a possibilidade de desvios do processamento, para outros pontos de fluxo, dependendo do resultado de comparações (se... senão). Exemplo: Verificar sexo se F... senão M.
	Conector - Indica a continuação do fluxo de processamento em outra parte do diagrama

Símbolos utilizados no diagrama de blocos (fluxograma)



Na Tabela a seguir é possível observar os símbolos utilizados para a entrada de dados.

	Operação manual (entrada de dados) - Utilizado para ler dados necessários ao programa fora de linha e sem a intervenção de dispositivos mecânicos.
	Teclado - Serão as informações recebidas pelo teclado

Símbolos utilizados na entrada de dados do diagrama de blocos

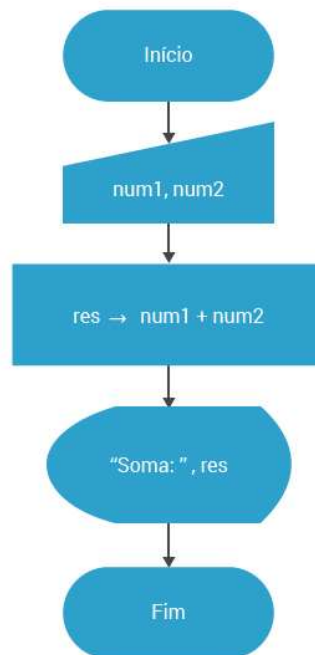


Na Tabela a seguir é possível observar os símbolos utilizados para a saída de dados.

	Display - Mostra o resultado pelo monitor
	Relatório em Impressora - É utilizado quando se deseja que os dados sejam impressos

Símbolos utilizados na saída de dados do diagrama de blocos

Dentro do símbolo sempre terá algo escrito, pois somente os símbolos não nos dizem nada. No exemplo a seguir é possível observar o diagrama de blocos para o algoritmo da soma dos dois números inteiros. É possível notar que, no símbolo de saída de dados, a mensagem a ser exibida é composta por texto (entre aspas) e pelo valor da variável.



Exemplo de diagrama de blocos para o algoritmo da soma de dois números

No exemplo a seguir, dado um número inteiro denominado como *v1*, é feito o cálculo do quadrado deste número e, sem seguida, o resultado é exibido. É possível notar que, no símbolo de saída de dados, a mensagem a ser exibida é composta somente pelo valor da variável.

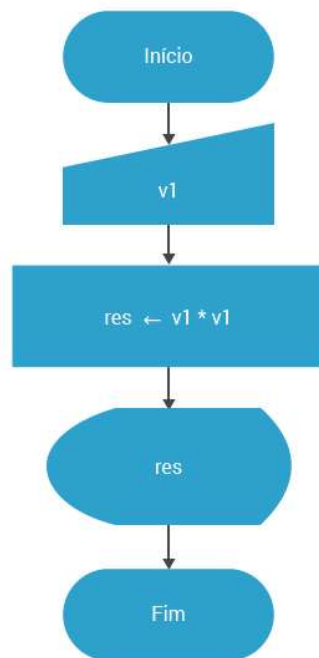


Diagrama de blocos que calcula o quadrado de um número inteiro

Como foi visto até agora, o diagrama de blocos é a primeira forma de notação gráfica, mas existe outra, que é uma técnica narrativa denominada pseudocódigo, também conhecida como português estruturado ou chamada por alguns de portugol.

Português Estruturado (Pseudocódigo ou Portugol)

Os algoritmos são descritos em uma linguagem chamada pseudocódigo. Para evitar problemas de ambiguidade utilizaremos um conjunto de regras que visam restringir e estruturar o uso do português na representação dos algoritmos e que, intencionalmente, se aproximam da maneira pela qual o fazem as linguagens de programação reais (como Pascal e C), com a finalidade de facilitar a futura codificação dos algoritmos. O português estruturado é um tipo de algoritmo que utiliza uma linguagem flexível, intermediária entre a linguagem natural e a linguagem de programação.

Assim, temos o Portugol ou Português Estruturado, que é uma pseudolinguagem de programação, que permite pensarmos no problema e não na máquina que vai executar o algoritmo.

As vantagens do português estruturado são:

- Usa o português como base.
- Passagem quase imediata do algoritmo para uma linguagem de programação qualquer.

As desvantagens do português estruturado são:

- Exige a definição de uma linguagem não real para trabalho.



A estrutura de um algoritmo representado em português estruturado é a seguinte:

```
1. algoritmo <nome_do_algoritmo>
2. var
3.     <Declaração das variáveis>
4. inicio
5.
6.     <Entrada de Dados>
7.
8.     <Processamento de Dados>
9.
10.    <Saída de Dados>
11.
12. fim
```

O mesmo algoritmo para a soma de dois números inteiros pode ser representado em português estruturado da seguinte forma:

```
1. algoritmo Soma_Numeros
2. var
3.     num1, num2, soma: inteiro
4. inicio
5.
6.     leia num1, num2
7.     soma <- num1 + num2
8.     escreva "Soma: ", soma
9.
10. fim
```

O algoritmo para o cálculo do quadrado de um número inteiro pode ser representado em português estruturado da seguinte forma:

```
1. algoritmo Calcula_Quadrado
2. var
3.     v1, quadrado: inteiro
4. inicio
5.
6.     leia v1
7.     quadrado <- v1 * v1
8.     escreva quadrado
9.
10. fim
```

No algoritmo da soma de dois números mostrado anteriormente, alguns comandos foram utilizados para a entrada, processamento e saída de dados.

Na fase de Entrada de Dados é utilizado o comando para a leitura dos dados denominado "leia". Os dados lidos por um algoritmo devem ser armazenados em variáveis. A sintaxe para a utilização do comando "leia" é a seguinte:

```
1. leia <variavel(is)>
```

A seguir, apresentamos alguns exemplos da utilização do comando "leia" para a leitura dos dados da variável "raio" e de três variáveis ("N1, N2 e N3"). É possível ler o valor de vários dados em uma única instrução, para isto, basta separar cada identificador de variável por vírgula.



1. leia raio
2. leia N1, N2, N3

Na Fase do Processamento de Dados um comando de atribuição permite-nos fornecer um valor a uma variável, em que o tipo deve ser compatível com o tipo da variável. Para que um valor seja atribuído a uma variável, devemos utilizar o símbolo "<-". A sintaxe para a utilização do comando de atribuição é a seguinte:

1. identificador da variavel <- expressao

A seguir, apresentamos alguns exemplos da utilização do comando de atribuição.

1. A <- B
2. X <- 10
3. Y <- C + D + 1
4. SOMA <- A1 + A2 + 6

Na fase de Saída de Dados é utilizado o comando para a escrita dos dados denominado "escreva". Os dados de saída de um algoritmo podem vir de variáveis, expressões ou constantes. A sintaxe para a utilização do comando "escreva" é a seguinte:

1. escreva <variavel(is) ou informacoes>

A seguir, apresentamos alguns exemplos da utilização do comando "escreva" para exibir simplesmente um texto ("Olá Mundo!"), uma variável (*res*), um texto com uma variável ("*Soma :*", *soma*) e duas variáveis (*N1* e *N2*). É possível exibir o valor de vários dados em uma única instrução, para isto, basta separar cada dado por vírgula.

1. escreva "Ola Mundo"
2. escreva res
3. escreva "Soma: ", soma
4. escreva N1, N2

Devido ao fato de o Português Estruturado ser a forma que mais se aproxima de uma linguagem de programação, ele será o mais utilizado para a representação dos algoritmos.

Quiz

Exercício Final



[INICIAR >](#)

Referências

FORBELLONE, A. L. V; EBERSPACHER, H. F. Lógica de Programação: a construção de algoritmos e estruturas de dados (3a. edição). São Paulo: Prentice Hall, 2005.

RISSETTI, G.; PUGA, S. Lógica de Programação e estruturas de dados, com aplicações em Java. São Paulo: Prentice Hall, 2003.

SOUZA, M. A. F. e outros. Algoritmos e lógica de programação: um texto introdutório para Engenharia (2a. edição). São Paulo: Cengage Learning, 2011.



Avalie este tópico



ANTERIOR

Conceitos iniciais: dados, tipos de dados, variáveis e constantes



Índice

Biblioteca

([https://www.uninove.br/conheca-](https://www.uninove.br/conheca-a-uninove/biblioteca/sobre-a-biblioteca/apresentacao/)

a-

uninove/biblioteca/sobre-

a-

biblioteca/apresentacao/)

Portal Uninove

(<http://www.uninove.br>)

Mapa do Site

Ajuda?

PRÓXIMO
([https://ava.un](https://ava.uninove.br/so/)

Introdução à linguagem C++

® Todos os direitos reservados

