

## Curso Completo de Algoritmos e Lógica de Programação

[educandoweb.com.br](http://educandoweb.com.br)

**Prof. Dr. Nelio Alves**

### Capítulo: Linguagem C++

#### Atenção

Este material foi elaborado para alunos do curso **Algoritmos e Lógica de Programação** do professor Nelio Alves.

A didática do conteúdo deste material está adaptada para quem já fez a parte do referido curso sobre construção de algoritmos na linguagem do VisualG.

Para mais informações:  
[educandoweb.com.br](http://educandoweb.com.br)

#### Instalação das ferramentas

IDE : Code Blocks

<http://www.codeblocks.org/downloads>

- Download the binary release
- Escolher pacote de instalação (setup) com **Code Blocks + Mingw**

#### Primeiro programa em C++

VisualG	C++
Algoritmo "primeiro"	<code>#include &lt;iostream&gt;</code>
Var	<code>using namespace std;</code>
Inicio	<code>int main()</code>
escreval("Ola mundo!")	<code>{</code>
Fimalgoritmo	<code>    cout &lt;&lt; "Ola mundo!" &lt;&lt; endl;</code>
	<code>    return 0;</code>
	<code>}</code>

## TIPOS DE DADOS E VARIÁVEIS

Significado	Tipo VisualG	Tipo C++	Valor padrão	Observação
número inteiro	inteiro	<b>int</b>	não atribuído (lixo de memória)	<b>int</b> : -32767 a 32767 ou -2147483648 a 2147483647 <b>long int</b> : -2147483648 a 2147483647 <b>long long int</b> : -9223372036854775807 a 9223372036854775807
número de ponto flutuante	real	<b>double</b>	não atribuído (lixo de memória)	<b>float</b> : precisão simples <b>double</b> : precisão dupla
um único caractere	caractere	<b>char</b>	não atribuído (lixo de memória)	Valores literais devem ter aspas simples. Exemplo: 'F'
texto	caractere	<b>string</b>	não atribuído (lixo de memória)	Valores literais devem ter aspas duplas. Exemplo: "Maria"
valor lógico	logico	<b>bool</b>	não atribuído (lixo de memória)	Valores possíveis: false / true

### Referência:

<https://pt.cppreference.com/w/cpp/language/types>

## Exemplo: declaração e atribuição de variáveis

VisualG	C++
<p>Algoritmo "teste"</p> <p>Var</p> <pre> idade : inteiro salario, altura : real genero : caractere nome : caractere </pre> <p>Inicio</p> <pre> idade &lt;- 20 salario &lt;- 5800.5 altura &lt;- 1.63 genero &lt;- "F" nome &lt;- "Maria Silva"  escreval("IDADE = ", idade) escreval("SALARIO = ", salario:4:2) escreval("ALTURA = ", altura:4:2) escreval("GENERO = ", genero) escreval("NOME = ", nome) </pre> <p>Fimalgoritmo</p>	<pre> #include &lt;iostream&gt; #include &lt;iomanip&gt; #include &lt;string&gt;  using namespace std;  int main() {      int idade;     double salario, altura;     char genero;     string nome;      idade = 20;     salario = 5800.5;     altura = 1.63;     genero = 'F';     nome = "Maria Silva";      cout &lt;&lt; fixed &lt;&lt; setprecision(2);     cout &lt;&lt; "IDADE = " &lt;&lt; idade &lt;&lt; endl;     cout &lt;&lt; "SALARIO = " &lt;&lt; salario &lt;&lt; endl;     cout &lt;&lt; "ALTURA = " &lt;&lt; altura &lt;&lt; endl;     cout &lt;&lt; "GENERO = " &lt;&lt; genero &lt;&lt; endl;     cout &lt;&lt; "NOME = " &lt;&lt; nome &lt;&lt; endl;      return 0; } </pre>

**NOTA:** A linguagem C++ aceita atribuição diretamente na declaração da variável, inclusive para texto. Por exemplo:

```

int idade = 20;
string nome = "Maria Silva";

```

## OPERADORES EM C++

### Operadores aritméticos

Operador	Significado
+	adição
-	subtração
*	multiplicação
/	divisão
%	resto da divisão ("mod")

### Operadores comparativos

Operador	Significado
<	menor
>	maior
<=	menor ou igual
>=	maior ou igual
==	igual
!=	diferente

### Operadores lógicos

Operador	Significado
&&	e
	ou
!	não

## SAÍDA DE DADOS EM C++

Comando no VisualG	Comando em C++	Biblioteca / namespace
escreva / escreval	cout <<	<code>#include &lt;iostream&gt;</code>  <code>using namespace std;</code>

Formatação de ponto flutuante	Biblioteca
cout << fixed << setprecision(2)	<code>#include &lt;iomanip&gt;</code>  <code>using namespace std;</code>

Biblioteca universal (inclui tudo e dispensa bibliotecas individuais)
<code>#include &lt;bits/stdc++.h&gt;</code>

## SAÍDA DE DADOS EM C++

Exemplo VisualG	Exemplo C++	Resultado na tela
<pre>escreva("Bom dia") escreva("Boa noite")</pre>	<pre>cout &lt;&lt; "Bom dia"; cout &lt;&lt; "Boa noite";</pre>	Bom diaBoa noite
<pre>escreval("Bom dia") escreval("Boa noite")</pre>	<pre>cout &lt;&lt; "Bom dia" &lt;&lt; endl; cout &lt;&lt; "Boa noite" &lt;&lt; endl;</pre>	Bom dia Boa noite
<pre>x, y : inteiro x &lt;- 10 y &lt;- 20 escreval(x) escreval(y)</pre>	<pre>int x, y; x = 10; y = 20; cout &lt;&lt; x &lt;&lt; endl; cout &lt;&lt; y &lt;&lt; endl;</pre>	10 20
<pre>x : real x &lt;- 2.3456 escreval(x:4:2)</pre>	<pre>double x; x = 2.3456; cout &lt;&lt; fixed &lt;&lt; setprecision(2) &lt;&lt; x &lt;&lt; endl;</pre>	2.35
<pre>idade : inteiro salario : real nome : caractere sexo : caractere  idade &lt;- 32 salario &lt;- 4560.9 nome &lt;- "Maria Silva" sexo &lt;- "F"  escreval("A funcionaria ", nome, ", sexo ", sexo, ", ganha ", salario:8:2, " e tem ", idade, " anos.")</pre>	<pre>int idade; double salario; string nome; char sexo;  idade = 32; salario = 4560.9; nome = "Maria Silva"; sexo = 'F';  cout &lt;&lt; fixed &lt;&lt; setprecision(2); cout &lt;&lt; "A funcionaria " &lt;&lt; nome &lt;&lt; ", sexo " &lt;&lt; sexo &lt;&lt; ", ganha " &lt;&lt; salario &lt;&lt; " e tem " &lt;&lt; idade &lt;&lt; " anos" &lt;&lt; endl;</pre>	A funcionaria Maria Silva, sexo F, ganha 4560.90 e tem 32 anos

## PROCESSAMENTO DE DADOS / CASTING EM C++

Exemplo VisualG	Exemplo C++	Resultado na tela
<pre>x, y : inteiro x &lt;- 5 y &lt;- 2 * x escreval(x) escreval(y)</pre>	<pre>int x, y; x = 5; y = 2 * x; cout &lt;&lt; x &lt;&lt; endl; cout &lt;&lt; y &lt;&lt; endl;</pre>	<pre>5 10</pre>
<pre>x : inteiro y : real x &lt;- 5 y &lt;- 2 * x escreval(x) escreval(y)</pre>	<pre>int x; double y; x = 5; y = 2 * x; cout &lt;&lt; x &lt;&lt; endl; cout &lt;&lt; fixed &lt;&lt; setprecision(1) &lt;&lt; y &lt;&lt; endl;</pre>	<pre>5 10.0</pre>
<pre>b1, b2, h, area : real b1 &lt;- 6.0 b2 &lt;- 8.0 h &lt;- 5.0 area &lt;- (b1 + b2) / 2.0 * h; escreval(area)</pre>	<pre>double b1, b2, h, area; b1 = 6.0; b2 = 8.0; h = 5.0; area = (b1 + b2) / 2.0 * h; cout &lt;&lt; area &lt;&lt; endl;</pre>	<pre>35</pre>
<pre>a, b, resultado : inteiro a &lt;- 5 b &lt;- 2 resultado &lt;- a \ b escreval(resultado)</pre>	<pre>int a, b, resultado; a = 5; b = 2; resultado = a / b; cout &lt;&lt; resultado &lt;&lt; endl;</pre>	<pre>2</pre>
<pre>a : real b : inteiro a &lt;- 5.0 b &lt;- Int(a) escreval(b)</pre>	<pre>double a; int b; a = 5.0; b = (int) a; cout &lt;&lt; b &lt;&lt; endl;</pre>	<pre>5</pre>

## ENTRADA DE DADOS EM C++

Comando no VisualG	Comando em C++	Biblioteca
leia	<code>cin &gt;&gt;</code>  <code>getline</code> (para ler texto até a quebra de linha)	<code>#include &lt;iostream&gt;</code>  <code>using namespace std;</code>

### Comando para limpeza de buffer

```
cin.ignore(INT_MAX, '\n');
```

**QUANDO USAR:** quando você for ler um texto até a quebra de linha (`getline`), mas antes o seu programa já leu algum outro dado e deixou uma quebra de linha pendente. Por exemplo:

```
cout << "Digite sua idade: ";
cin >> idade;
cout << "Digite seu nome completo: ";
cin.ignore(INT_MAX, '\n');    // ----- limpeza de buffer
getline(cin, nome);
```



**ENTRADA DE DADOS EM C++**

Exemplo VisualG	Exemplo C++
<p>Algoritmo "teste_entrada"</p> <p>Var</p> <p>    salario1, salario2 : real          nome1, nome2 : caractere          idade : inteiro          sexo : caractere</p> <p>Inicio</p> <p>    escreva("Nome da primeira pessoa: ")          leia(nome1)          escreva("Salario da primeira pessoa: ")          leia(salario1)</p> <p>    escreva("Nome da segunda pessoa: ")          leia(nome2)          escreva("Salario da segunda pessoa: ")          leia(salario2)</p> <p>    escreva("Digite uma idade: ")          leia(idade)          escreva("Digite um sexo (F/M): ")          leia(sexo)</p> <p>    escreval("Nome 1: ", nome1)          escreval("Salario 1: ", salario1:4:2)          escreval("Nome 2: ", nome2)          escreval("Salario 2: ", salario2:4:2)          escreval("Idade: ", idade)          escreval("Sexo: ", sexo)</p> <p>Fimalgoritmo</p>	<pre>#include &lt;iostream&gt; #include &lt;iomanip&gt; #include &lt;string&gt; #include &lt;climits&gt;  using namespace std;  int main() {     double salario1, salario2;     string nome1, nome2;     int idade;     char sexo;      cout &lt;&lt; "Nome da primeira pessoa: ";     getline(cin, nome1);     cout &lt;&lt; "Salario da primeira pessoa: ";     cin &gt;&gt; salario1;      cout &lt;&lt; "Nome da segunda pessoa: ";     cin.ignore(INT_MAX, '\n'); // ----- limpeza de buffer     getline(cin, nome2);     cout &lt;&lt; "Salario da segunda pessoa: ";     cin &gt;&gt; salario2;      cout &lt;&lt; "Digite uma idade: ";     cin &gt;&gt; idade;     cout &lt;&lt; "Digite um sexo (F/M): ";     cin &gt;&gt; sexo;      cout &lt;&lt; fixed &lt;&lt; setprecision(2);     cout &lt;&lt; "Nome 1: " &lt;&lt; nome1 &lt;&lt; endl;     cout &lt;&lt; "Salario 1: " &lt;&lt; salario1 &lt;&lt; endl;     cout &lt;&lt; "Nome 2: " &lt;&lt; nome2 &lt;&lt; endl;     cout &lt;&lt; "Salario 2: " &lt;&lt; salario2 &lt;&lt; endl;     cout &lt;&lt; "Idade: " &lt;&lt; idade &lt;&lt; endl;     cout &lt;&lt; "Sexo: " &lt;&lt; sexo &lt;&lt; endl;      return 0; }</pre>

## **COMO EXECUTAR O DEBUGGER NO CODE BLOCKS**

**ATENÇÃO:** o debug NÃO FUNCIONA para arquivos isolados. Seu programa deve estar dentro de um PROJETO.

### **PASSOS PARA CONFIGURAR O DEBUGGER:**

- Settings -> Compiler -> Toolchain Executables
  - Debugger: GDB/CDB debugger: default
- Settings -> Debugger -> GDB/CDB debugger -> Default
  - Executable path: C:\Program Files (x86)\CodeBlocks\MinGW\bin\gdb32.exe

### **COMANDOS DO DEBUGGER:**

- Habilitar/desabilitar breakpoint: **F5**
- Iniciar o debug: **F8**
- Rodar um passo: **F7**
- Parar o debug: **SHIFT+F8**
- Mostrar variáveis: **Debug -> Debugging windows -> Watches**

## ESTRUTURA CONDICIONAL EM C++

Simple	Composta	Encadeamento
<pre>if (condição) {     comando1     comando2 }</pre>	<pre>if (condição) {     comando1     comando2 } else {     comando3     comando4 }</pre>	<pre>if (condição1) {     comando1     comando2 } else if (condição2) {     comando3     comando4 } else {     comando5     comando6 }</pre>

Exemplo VisualG	Exemplo C++
<p>Algoritmo "teste_condicional"</p> <p>Var</p> <p>    hora : inteiro</p> <p>Inicio</p> <p>    escreva("Digite uma hora do dia: ")</p> <p>    leia(hora)</p> <p>    se hora &lt; 12 entao</p> <p>        escreval("Bom dia!")</p> <p>    senao</p> <p>        escreval("Boa tarde!")</p> <p>    fimse</p> <p>Fimalgoritmo</p>	<pre>#include &lt;iostream&gt; using namespace std; int main() {     int hora;     cout &lt;&lt; "Digite uma hora do dia: ";     cin &gt;&gt; hora;      if (hora &lt; 12) {         cout &lt;&lt; "Bom dia!" &lt;&lt; endl;     }     else {         cout &lt;&lt; "Boa tarde!" &lt;&lt; endl;     }      return 0; }</pre>

## ESTRUTURA ENQUANTO EM C++

Sintaxe	Regra
<pre>while (<b>condição</b>) {     <b>comando1</b>     <b>comando2</b> }</pre>	<p>V: executa e volta F: pula fora</p>

Exemplo VisualG	Exemplo C++
<p>Algoritmo "teste_enquanto"</p> <p>Var</p> <p>    x, soma : inteiro</p> <p>Inicio</p> <p>    soma &lt;- 0</p> <p>    escreva("Digite o primeiro numero: ")</p> <p>    leia(x)</p> <p>    enquanto x &lt;&gt; 0 faca</p> <p>        soma &lt;- soma + x</p> <p>        escreva("Digite outro numero: ")</p> <p>        leia(x)</p> <p>    fimenquanto</p> <p>    escreval("SOMA = ", soma)</p> <p>Fimalgoritmo</p>	<pre>#include &lt;iostream&gt;  using namespace std;  int main() {      int x, soma;      soma = 0;     cout &lt;&lt; "Digite o primeiro numero: ";     cin &gt;&gt; x;      while (x != 0) {         soma = soma + x;         cout &lt;&lt; "Digite outro numero: ";         cin &gt;&gt; x;     }      cout &lt;&lt; "SOMA = " &lt;&lt; soma;      return 0; }</pre>

## ESTRUTURA PARA EM C++

### Sintaxe / regra

Executa somente  
na primeira vez

V: executa e volta  
F: pula fora

Executa toda vez depois  
de voltar

```
for ( início ; condição ; incremento ) {
    comando1
    comando2
}
```

### Exemplo VisualG

Algoritmo "teste\_para"

Var

N, i, x, soma : inteiro

Inicio

escreva("Quantos numeros serao digitados? ")  
leia(N)

soma <- 0  
para i de 1 ate N faca  
    escreva("Digite um numero: ")  
    leia(x)  
    soma <- soma + x  
fimpara

escreval("SOMA = ", soma)

Fimalgoritmo

### Exemplo C++

```
#include <iostream>
using namespace std;
int main() {
    int N, i, x, soma;
    cout << "Quantos numeros serao digitados? ";
    cin >> N;

    soma = 0;
    for (i = 1; i <= N; i++) {
        cout << "Digite um numero: ";
        cin >> x;
        soma = soma + x;
    }

    cout << "SOMA = " << soma << endl;

    return 0;
}
```

## ESTRUTURA DO-WHILE EM C++ (variante da estrutura REPITA-ATÉ)

Sintaxe	Regra
<pre>do {     comando 1     comando 2 } while (condição);</pre>	<p>V: volta F: pula fora</p>

Exemplo VisualG	Exemplo C++
<p>Algoritmo "exemplo_repita_ate"</p> <p>Var</p> <p>    C, F : real</p> <p>    resp : caractere</p> <p>Inicio</p> <p>    repita</p> <p>        escreva("Digite a temperatura em Celsius: ")</p> <p>        leia(C)</p> <p>        F &lt;- 9.0 * C / 5.0 + 32.0</p> <p>        escreval("Equivalente em Fahrenheit: ", F:6:1)</p> <p>        escreva("Deseja repetir (s/n)? ")</p> <p>        leia(resp)</p> <p>    ate resp &lt;&gt; "s"</p> <p>Fimalgoritmo</p>	<pre>#include &lt;bits/stdc++.h&gt;  using namespace std;  int main() {     double C, F;     char resp;      do {         cout &lt;&lt; "Digite a temperatura em Celsius: ";         cin &gt;&gt; C;         F = 9.0 * C / 5.0 + 32.0;         cout &lt;&lt; "Equivalente em Fahrenheit: " &lt;&lt; F &lt;&lt; endl;         cout &lt;&lt; "Deseja repetir (s/n)? ";         cin &gt;&gt; resp;     } while (resp == 's');      return 0; }</pre>

## VETORES EM C++

Exemplo VisualG	Exemplo C++
<p>Algoritmo "teste_vetor"</p> <p>Var</p> <p>vet: vetor [0..9] de real N, i : inteiro</p> <p>Inicio</p> <p>escreva("Quantos numeros voce vai digitar? ") leia(N)</p> <p>para i de 0 ate N-1 faca   escreva("Digite um numero: ")   leia(vet[i]) fimpara</p> <p>escreval escreval("NUMEROS DIGITADOS:") para i de 0 ate N-1 faca   escreval(vet[i]:8:1) fimpara</p> <p>Fimalgoritmo</p>	<pre>#include &lt;iostream&gt; #include &lt;iomanip&gt;  using namespace std;  int main() {      int N, i;      cout &lt;&lt; "Quantos numeros voce vai digitar? ";     cin &gt;&gt; N;      double vet[N];      for (i = 0; i &lt; N; i++) {         cout &lt;&lt; "Digite um numero: ";         cin &gt;&gt; vet[i];     }      cout &lt;&lt; endl &lt;&lt; "NUMEROS DIGITADOS:" &lt;&lt; endl;     cout &lt;&lt; fixed &lt;&lt; setprecision(1);     for (i = 0; i &lt; N; i++) {         cout &lt;&lt; vet[i] &lt;&lt; endl;     }      return 0; }</pre>

## MATRIZES EM C++

Exemplo VisualG	Exemplo C++
<p>Algoritmo "teste_matriz"</p> <p>Var</p> <p>mat: vetor [0..4, 0..4] de inteiro</p> <p>M, N, i, j : inteiro</p> <p>Inicio</p> <p>  escreva("Quantas linhas vai ter a matriz? ")</p> <p>  leia(M)</p> <p>  escreva("Quantas colunas vai ter a matriz? ")</p> <p>  leia(N)</p> <p>  para i de 0 ate M-1 faca</p> <p>    para j de 0 ate N-1 faca</p> <p>      escreva("Elemento [", i, ",", j, "]: ")</p> <p>      leia(mat[i, j])</p> <p>    fimpara</p> <p>  fimpara</p> <p>  escreval</p> <p>  escreval("MATRIZ DIGITADA:")</p> <p>  para i de 0 ate M-1 faca</p> <p>    para j de 0 ate N-1 faca</p> <p>      escreva(mat[i, j])</p> <p>    fimpara</p> <p>  escreval</p> <p>  fimpara</p> <p>Fimalgoritmo</p>	<pre>#include &lt;iostream&gt;  using namespace std;  int main() {      int M, N, i, j;      cout &lt;&lt; "Quantas linhas vai ter a matriz? ";     cin &gt;&gt; M;     cout &lt;&lt; "Quantas colunas vai ter a matriz? ";     cin &gt;&gt; N;      int mat[M][N];      for (i = 0; i &lt; M; i++) {         for (j = 0; j &lt; N; j++) {             cout &lt;&lt; "Elemento [" &lt;&lt; i &lt;&lt; "," &lt;&lt; j &lt;&lt; "]: ";             cin &gt;&gt; mat[i][j];         }     }      cout &lt;&lt; endl &lt;&lt; "MATRIZ DIGITADA:" &lt;&lt; endl;     for (i = 0; i &lt; M; i++) {         for (j = 0; j &lt; N; j++) {             cout &lt;&lt; mat[i][j] &lt;&lt; " ";         }         cout &lt;&lt; endl;     }      return 0; }</pre>