
Alo mundo!

Entradas e Saídas em C/C++

Prof. Karl Apaza Agüero

Estrutura Geral de um Programa em C

```
#include <bibliotecas>
#define <constantes>
<variáveis globais>

int main ([parâmetros]) {
    <variáveis locais>
    <instruções>
    return 0;
}
```

Alo mundo em C

alo_mundo.c

Descrição

Seu programa deve cumprimentar o mundo.

Entrada

Este programa não possui entrada.

Saída

Seu programa deve imprimir a sentença *Alo mundo!* seguida de uma quebra de linha.

Alo mundo!

Código:

```
Linha 1    #include <stdio.h>
Linha 2    int main()
Linha 3    {
Linha 4        printf("Alo mundo!\n");
Linha 5        return 0;
Linha 6    }
```

Alo mundo!

Código:

```
Linha 1    #include <stdio.h>  
Linha 2    int main()  
Linha 3    {  
Linha 4        printf("Alo mundo!\n");  
Linha 5        return 0;  
Linha 6    }
```

Linha 1: avisa ao compilador que irá usar funções de entrada e saída de dados guardadas na biblioteca stdio. Neste caso a função usada é printf.

Alo mundo!

Código:

```
Linha 1    #include <stdio.h>
Linha 2    int main()
Linha 3    {
Linha 4        printf("Alo mundo!\n");
Linha 5        return 0;
Linha 6    }
```

Linha 2: Função main que todo programa C deve conter, pois é nesta função que o programa obrigatoriamente começa sua execução.

Alo mundo!

Código:

```
Linha 1    #include <stdio.h>
Linha 2    int main()
Linha 3    {
Linha 4        printf("Alo mundo!\n");
Linha 5        return 0;
Linha 6    }
```

Linha 2: A função main vai retornar um valor inteiro (int) ao final de sua execução e não vai precisar receber nenhum argumento para sua execução.

Alo mundo!

Código:

```
Linha 1    #include <stdio.h>
Linha 2    int main()
Linha 3    {
Linha 4        printf("Alo mundo!\n");
Linha 5        return 0;
Linha 6    }
```

Linhas 3 e 6: As chaves marcam o início e fim da função main.

Alo mundo!

Código:

```
Linha 1    #include <stdio.h>
Linha 2    int main()
Linha 3    {
Linha 4    printf("Alo mundo!\n");
Linha 5    return 0;
Linha 6    }
```

Linha 4: Para imprimir o texto *Alo mundo!* o programa usa a função printf. O início e o fim do texto são marcados pelo caractere ". O símbolo `\n` permite fazer a quebra de linha ao final do texto.

Alo mundo!

Código:

```
Linha 1    #include <stdio.h>
Linha 2    int main()
Linha 3    {
Linha 4        printf("Alo mundo!\n");
Linha 5        return 0;
Linha 6    }
```

Linha 5: A função termina com o comando return 0, que avisa ao sistema operacional, que foi quem iniciou a execução do programa, que o programa terminou sem problemas.

Alo mundo em C++

alo_mundo.cpp

Descrição

Seu programa deve cumprimentar o mundo.

Entrada

Este programa não possui entrada.

Saída

Seu programa deve imprimir a sentença *Alo mundo!* seguida de uma quebra de linha.

Alo mundo em C++

Código:

```
Linha 1    #include <iostream>
Linha 2    using namespace std;
Linha 3    int main()
Linha 4    {
Linha 5        cout<<"Alo mundo!"<<endl;
Linha 6        return 0;
Linha 7    }
```

Exercício

Descrição

Fazer um programa em C e C++ para mostrar uma mensagem com 3 linhas.

Entrada

Este programa não possui entrada.

Saída

Seu programa deve imprimir a mensagem a seguir:

Linha 1

Linha 2

Linha 3

Exercício em C

Código 1:

```
#include <stdio.h>

int main() {
    printf("Linha 1\n");
    printf("Linha 2\n");
    printf("Linha 3\n");
    return 0;
}
```

Código 2:

```
#include <stdio.h>

int main() {
    printf("Linha 1\nLinha 2\nLinha 3\n");
    return 0;
}
```

Exercício em C++

Código 1:

```
#include <iostream>
using namespace std;
int main() {
    cout<<"Linha 1"<<endl;
    cout<<"Linha 2"<<endl;
    cout<<"Linha 3"<<endl;
    return 0;
}
```

Código 2:

```
#include <iostream>
using namespace std;
int main() {
    cout<<"Linha 1"<<endl<<"Linha
    2"<<endl<<"Linha 3"<<endl;
    return 0;
}
```

Entradas e saídas básicas em C

- ▶ Entrada em C: função **scanf**

- ▶ A função **scanf** é utilizada para a entrada de dados a partir do teclado. Precisa da inclusão da biblioteca **stdio.h**. Exemplo:

```
#include <stdio.h>
```

```
int numero;  
scanf("%d", &numero);
```

- ▶ Para cada valor que se deseja ler, deve existir um especificador de formato, colocado dentro de uma string de formatação.
-

Entradas e saídas básicas em C

Especificadores de formato

Especificador de formato	Tipo de dado
%c	Caractere (char)
%d	Inteiro (int)
%o	inteiro em formato octal
%X, %x	inteiro em formato hexadecimal
%u	Inteiro sem sinal (unsigned int)
%ld	Inteiro longo (long int)
%f	Real ou ponto flutuante (float ou double)
%e, %E	Real ou ponto flutuante em formato exponencial
%g, %G	Real ou ponto flutuante. C escolhe melhor maneira de exibição entre normal e exponencial
%s	Cadeia de caracteres (strings) (char *)
%p	Endereço de um ponteiro

Entradas e saídas básicas em C

- ▶ Saída em C: função **printf**
 - ▶ A função **printf** faz com que os dados sejam escritos na tela do computador. Precisa da inclusão da biblioteca **stdio.h**. Exemplo:

```
#include <stdio.h>
```

```
int numero=10;  
printf("%d\n", numero);
```

- ▶ Para cada valor que se deseja imprimir, deve existir um especificador de formato. A string de formatação pode ser finalizada com o caractere de nova linha **\n**.
-

Entradas e saídas básicas em C

Especificadores de formato

Especificador de formato	Tipo de dado
%c	Caractere (char)
%d	Inteiro (int)
%o	inteiro em formato octal
%x, %X	inteiro em formato hexadecimal
%u	Inteiro sem sinal (unsigned int)
%ld	Inteiro longo (long int)
%f	Real ou ponto flutuante (float ou double)
%e, %E	Real ou ponto flutuante em formato exponencial
%g, %G	Real ou ponto flutuante. C escolhe melhor maneira de exibição entre normal e exponencial
%s	Cadeia de caracteres (strings) (char *)
%p	Endereço de um ponteiro

Entradas e saídas básicas em C

Caracteres de Escape

Caracteres de escape	Significado
<code>\n</code>	Nova linha
<code>\t</code>	Tabulação horizontal
<code>\v</code>	Tabulação vertical
<code>\\</code>	Caractere de barra invertida “\”
<code>\'</code>	Apóstrofe
<code>\"</code>	Aspas
<code>\?</code>	Interrogação
<code>\nnn</code>	Valor ASCII em octal
<code>\xnnn</code>	Valor ASCII em hexadecimal
<code>\a</code>	Aviso sonoro
<code>\b</code>	Retrocesso
<code>\f</code>	Avanço de formulário
<code>\r</code>	Retorno do carro (sem alimentação de linha)

Exemplo: Somar dois números em C

Descrição

Fazer um programa em C para somar dois números inteiros.

Entrada

Dois números inteiros.

Saída

Soma dos valores de entrada.

Exemplo: Somar dois números em C

Código:

```
#include <stdio.h>
```

```
int main() {  
    int a, b;  
    scanf("%d%d", &a, &b);  
    printf("%d\n", a+b);  
    return 0;  
}
```

Entradas e saídas básicas em C++

► Entrada: objeto **cin**

- O objeto **cin** é utilizado para a entrada de dados a partir do teclado. Precisa da inclusão da biblioteca **iostream** e do namespace **std**. Exemplo:

```
#include <iostream>
using namespace std;
```

```
int numero;
cin>>numero;
```

- Usa o operador de fluxo >> que realiza a leitura de uma sequência de dados, sem espaços e sem tabulações, vindas do teclado.
 - Namespaces são usados para evitar conflitos de nomes iguais que podem ocorrer quando o programa inclui várias bibliotecas.
-

Entradas e saídas básicas em C++

- ▶ Saída: objeto **cout**

- ▶ O objeto **cout** faz com que os dados sejam escritos na tela do computador. Precisa da inclusão da biblioteca **iostream** e o namespace **std**. Exemplo:

```
#include <iostream>
using namespace std;
```

```
int numero=10;
cout<<numero<<endl;
```

- ▶ Usa o operador de fluxo << que realiza a impressão dos dados na tela.
-

Exemplo: Somar dois números em C++

Descrição

Fazer um programa em C++ para somar dois números inteiros.

Entrada

Dois números inteiros.

Saída

Soma dos valores de entrada.

Exemplo: Somar dois números em C++

Código:

```
#include <iostream>
using namespace std;
int main() {
    int a, b;
    cin>>a>>b;
    cout<<a+b<<endl;
    return 0;
}
```

Exercício: Média

Descrição

Fazer um programa em C e C++ para calcular a média de 3 notas decimais de um aluno.

Entrada

3 notas do aluno.

Saída

Média do aluno.

Exercício: Média em C

Código:

```
#include <stdio.h>
```

```
int main() {  
    float a, b, c;  
    scanf("%f%f%f", &a, &b, &c);  
    printf("%f\n", (a+b+c)/3);  
    return 0;  
}
```

Exercício: Média em C++

Código:

```
#include <iostream>
using namespace std;
int main() {
    float a, b, c;
    cin>>a>>b>>c;
    cout<<(a+b+c)/3<<endl;
    return 0;
}
```
