

1997 - Hackers Iniciantes em Linguagem C

1997 - RoyalBug - thedarkage@mail.geocities.com
thedarkage2@mail.geocities.com

1997 - The Black Church MANUAL C 01

=====

1997 - <http://www.pcs.usp.br/~peasilva/>

1. PROLOGO

=====

Este manual foi desenvolvido para iniciantes em programação em C e C++.

Quem já conhece alguma linguagem de programação não-visual terá facilidade de com C e C++. Primeiramente irei ensinar C; todo código neste manual terá uma indicação se está em C, C++ ou utiliza recursos de ambos.

Para pegar o #2 deste manual entre em algum destes sites oficiais:

<http://www.pcs.usp.br/~peasilva/>

<http://www.geocities.com/SiliconValley/Way/5441/>

<http://www.geocities.com/SiliconValley/Bay/4885/>

2. Secoes

=====

Nesta seção, descrevo cada uma das outras seções deste manual:

3. Compiladores

Descrição dos principais compiladores C/C++ para DOS/WINDOWS/UNIX.

4. Programa: Hello World!

Ensina a utilizar saída de dados e variáveis. C

5. Programa: Hello Number!

Ensina a usar entrada de dados. C

6. Programa: Calculadora

Mostra como fazer cálculos. C

7. Tabela: printf

Tabela de argumentos possíveis para a função printf.

8. Algoritmo: Lendo arquivos

Código para ler um arquivo inteiro. Dificuldade (0-20): 0

9. Dica Rápida

10. Próxima edição

11. Criador/Editor/Colaboradores

As seções deste "curso" tem um modelo pre-definido:

1. Introdução ao curso

2. Descrição das seções do arquivo

3. Material sobre alguma coisa
4. Programa Exemplo
5. Programa Exemplo
6. Programa Exemplo
7. Tabela sobre alguma coisa
8. Algoritmo
9. Dica Rapida
10. Na proxima edicao
11. Criador/Editor/Colaboradores

3. Compiladores

=====

Para se rodar um programa em linguagem C/C++ é necessário um compilador. Aqui esta uma lista dos que conheço:

- Borland C++ 5.0 - Windows/Dos - Bom compilador, utilizo sua versao 4
- Watcom 11.0 - Dos - Roda seus programas em modo protegido. Nunca utilizei.
- Turbo C++ - Windows - Nao sei sua versao atual. Testei a 1.0. Nenhuma novidade.
- GNU - UNIX - Otimo compilador para UNIX. De graca! Procure-o na NET!
- DJGPP - DOS - Versao do GNU para DOS. De graca! Utilizo ele em meus programas.

Como compilar seu programa:

- Borland C++ - Digite no DOS: "bcc programa.xxx"
- DJGPP - Digite no DOS:
"gcc programa.c -s -oprograma.exe" - programas em c
"gxx programa.cc -s -oprograma.exe" - programas em c++

4. Programa: Hello World!

=====

Este programa ensina a usar variaveis e a colocar dados na tela.

```
<-----Inicio----->
/* Programa em C */
#include <stdio.h>
int main(){
    int inteiro;
    float longo=2.5;
    char caracter;
    inteiro=1;
    caracter='S';
    printf("Inteiro: %d\n",inteiro);
```

```

printf("Longo: %f\n",longo);
printf("Caracter: %c\n",caracter);
return(0);
}

```

<----- Fim ----->

O código anterior está em C e deve ser compilado como tal.

A primeira linha é um comentário. Os comentários em C devem **SEMPRE** estar entre `/*` e `*/`.

A segunda linha inclui o arquivo `"stdio.h"` no programa. Este arquivo é o arquivo básico para operações em C.

A terceira linha diz que a função `"main"` vai começar. A função `main` é a função que é executada quando o programa é executado. A palavra `"int"` que vem antes dela significa que ela retorna um número inteiro.

A quarta, quinta e sexta linha indicam que existem certas variáveis que devem existir. Na quarta, a variável `"inteiro"` é declarada como `"int"`, isto é, um número inteiro. Na quinta linha a variável `"longo"` é declarada como `"float"`, isto é, um número racional. Você pode perceber o `"=2.5"` que significa que a variável `"longo"` terá o valor `"2.5"`. Na sexta linha a variável `caracter` é criada como tipo `"character": caracter`. A sétima e a oitava linha setam as variáveis `"inteiro"` e `"caracter"` para `"1"` e `"S"` respectivamente.

Na nona linha, é usada a função `"printf"`. A função `"printf"` está no arquivo `"stdio.h"` que foi incluído no nosso arquivo na primeira linha. A função `printf` produz impressão formatada para a tela. O texto a ser impresso está entre as aspas: `"Inteiro: %d\n"`. Que exótico né?

Não. É normal. O `"\n"` significa nova linha. O `"%d"` significa coloca um inteiro na impressão. No caso, o valor de `"inteiro"` será impresso, preste atenção: `"printf("Inteiro: %d\n",inteiro);"`.

```

-----
|
|

```

A décima linha imprime um texto com uma variável `"float"` (`"%f"`).

A décima primeira linha imprime uma variável `"character"` (`"%c"`).

A décima segunda linha acaba o programa, retornando 0 na função `"main"`.

Este retorno (retorno de 0) significa que tudo deu certo. Caso tivesse ocorrido algum erro, devia-se retornar outro número qualquer.

Como você já deve ter visto, toda linha deve ser terminada com um `";"`, exceto as com `"{"` e `"}"` que significam onde uma instrução começa e termina (de uma olhada onde a função `main` começa e termina!).

Saída do programa:

Inteiro: 1

Longo: 2.5

Caracter: S

5. Programa: Hello Number!

=====

Este programa ensinara voce a obter dados do usuario.

```
<-----Inicio----->
/* Programa em C: hellonr.c */
#include <stdio.h>
int main(){
    int num1=0;
    printf("Numero 1: %d\n",num1);
    scanf("%d",&num1);
    printf("Novo numero: %d\n",num1);
    return(0);
}
<----- Fim ----->
```

Nas 3 primeiras linhas nao ha nenhuma novidade. Na linha 4, o programa inicia uma variavel num1 com valor 0. Na linha 5, ele mostra o valor de num1. Na linha 6 esta uma novidade. A funcao scanf pega do usuario um valor qualquer (verifique o %d, igual ao do printf!). A variavel que vai ser gravada a resposta esta logo depois: num1. "O que eh aquele &?". Aquilo significa que nao eh o valor que eu estou dando, e sim o endereco da variavel na memoria. Eh necessario fazer isso nesta funcao. O valor sera retornado para num1 e impresso na tela novamente (linha 7).

Saida do programa:

Numero 1: 0

15

Novo numero: 15

6. Programa: Calculadora

=====

Mais um programa galera. Este aqui faz calculos matematicos.

```
<-----Inicio----->
/* Programa em C: calc.c */
#include <stdio.h>
int main(){
    int num1=0, num2=0, num3=0,num4=0;
    int soma=0;
    int subtracao=0;
    int multiplicacao=0;
    int divisao=0;
```

```

int resto=0;
printf("Numero 1: ");
scanf("%d",&num1);
printf("Numero 2: ");
scanf("%d",&num2);
printf("Numero 3: ");
scanf("%d",&num3);
printf("Numero 4: ");
scanf("%d",&num4);
/* Calculando */
soma=num1+num2;
subtracao=num2-num3;
multiplicacao=num3*num4;
divisao=num4/num1;
resto=num4%num1;
printf("%d+%d=%d\n",num1,num2,soma);
printf("%d-%d=%d\n",num2,num3,subtracao);
printf("%d*%d=%d\n",num3,num4,multiplicacao);
printf("%d/%d=\t%d\n",num4,num1,divisao);
printf("Resto=\t%d\n",num4,num1,resto);
return(0);
}
<----- Fim ----->

```

Novidades:

Linha 4: Declaramos 4 variaveis em uma so linha! Declare quantas voce quiser!

Linha 19 ate 23: Fazemos calculos matematicos com +,-,*, / (dividido) e % (resto de divisao). O % eh usado para calcular o resto.

Linha 27: Usamos o comando "\t" que significa TAB.

7. Tabela: printf

=====

Simbolo	Significado
\n	Nova linha
\t	TAB
\b	Backspace
\\	\
\/	/
\"	"
%d	integer
%f	float
%c	cacacter

```
%s      string
%%      %
%lf     double
```

8. Algoritmo: Lendo Arquivos

=====

Este código ensina a ler um arquivo até o fim.

ATENÇÃO: Você precisa conhecer estas funções e estruturas!

```
<-----Inicio----->
while(!feof(arq)){
    fgets(char,1000,arq);
    /* faz tudo */
}
<----- Fim ----->
```

Substitua a palavra arq pela stream do seu arquivo de leitura.
Serão lidos até 1000 caracteres de uma linha e rodado o faz tudo.
Substitua o /* faz tudo */ pelo código que você quiser.

9. Dica Rápida

=====

Como colocar comentários em uma linha sem usar o "/* */":

Use o seguinte:

```
<-----Inicio----->
comece seu programa aqui
// Comentário
continue seu programa aqui
<----- Fim ----->
```

Em C, o comentário está sempre entre /* e */.

Em C++ o comentário está depois de //, sendo assim, o compilador ignora tudo que vem depois dele nesta linha.

10. Próxima edição

=====

Seções da próxima edição:

1. Introdução
2. Seções
3. Usando gráficos em C
4. Programa: Interpretador
5. Programa: Visualizador de Arquivos

6. Especial: WordMake.Exe - Criador de WordLists
7. Tabela: Nao definida
8. Algoritmo: Numeros Aleatorios
9. Dica Rapida
10. Proxima Edicao
11. Criador/Editor/Colaboradores

Aqui estao os materiais das proximas edicoes:

- Criando arquivos de cabecalho (.H)
- Criando bibliotecas (.A)
- WordLists
- Matrizes
- Mouse em C/C++
- Basico de C++
- Basico de Allegro
- Biblioteca de DEBUG
- Biblioteca de HighScore

11. Criador/Editor/Colaboradores

=====

Criador: RoyalBug - thedarkage@geocities.com
Editor: RoyalBug - thedarkage@geocities.com
Colaborador: Sacanagem! Ninguem!

1997 - RoyalBug - thedarkage@geocities.com

1997 - HardLoKi - peasilva@pcs.usp.br
peduardo@hotmail.com

1997 - The Black Church

THKS