

Funções de Entrada e Saída Formatada

➡ scanf ()

➡ leitura de dados;

➡ formato geral:

scanf (string_de_controle, lista_de_argumentos);

➡ *string_de_controle* → descrição de todas as variáveis que serão lidas, com informações de seus tipos e ordem em que serão lidas.

➡ *lista_de_argumentos* → lista com os identificadores das variáveis que serão lidas, em ordem compatível com a *string_de_controle*;

Funções de Entrada e Saída Formatada

➡ scanf (continuação)

Observação: colocar antes de cada identificador da *lista_de_argumentos* o caractere ‘&’

Exemplo: char ch;
 scanf ("%c", &ch);

Funções de Entrada e Saída Formatada

➤ scanf (continuação)

Tabela simplificada de códigos de formato (%)

Código	Formato
%c	Um caractere (char)
%d	Um número inteiro decimal (int)
%f	Ponto flutuante decimal
%s	String

Exercício:

Construa um programa em C que receba três notas e seus respectivos pesos, calcule e mostre a média ponderada dessas notas.

```

#include <stdio.h>
main ()
{
    float n1, n2, n3;
    int p1, p2, p3;
    printf("Digite a primeira nota: ");
    scanf("%f", &n1);
    printf("\nDigite o peso da primeira nota: ");
    scanf("%d", &p1);
    printf("Digite a segunda nota: ");
    scanf("%f", &n2);
    printf("\nDigite o peso da segunda nota: ");
    scanf("%d", &p2);
    printf("Digite a terceira nota: ");
    scanf("%f", &n3);
    printf("\nDigite o peso da terceira nota: ");
    scanf("%d", &p3);
    printf("\nA media ponderada eh %.2f",
    (n1*p1+n2*p2+n3*p3)/(p1+p2+p3));
}

```

Modificadores de Tipos

Palavras reservadas:

- **signed** (com sinal)
- **unsigned** (sem sinal)
- **long** (expande faixa de valores)
- **short** (contraí faixa de valores)

Exemplo: `unsigned char letra;`
 `long int numero1, numero2;`

Tipos de dados definidos no padrão ANSI

Tipo	Tamanho aproximado em bits	Faixa mínima
char	8	-127 a 127
unsigned char	8	0 a 255
signed char	8	-127 a 127
int	16	-32.767 a 32.767
unsigned int	16	0 a 65.535
signed int	16	O mesmo que int
short int	16	O mesmo que int
unsigned short int	16	0 a 65.535
signed short int	16	O mesmo que short int
long int	32	-2.147.483.647 a 2.147.483.647
signed long int	32	O mesmo que long int
unsigned long int	32	0 a 4.294.967.295
float	32	Seis dígitos de precisão
double	64	Dez dígitos de precisão
long double	80	Dez dígitos de precisão

Fonte: livro “C Completo e Total”

Tipos

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

```
main()
```

```
{
```

```
    char c;
```

```
    int i;
```

```
    short int si;
```

```
    unsigned int ui;
```

```
    long int li;
```

```
    float f;
```

```
    double d;
```

```
    printf("char %d \n",sizeof(c));
```

```
    printf("int %d \n",sizeof(i));
```

```
    printf("short int %d \n",sizeof(si));
```

```
    printf("unsigned int %d \n",sizeof(ui));
```

```
    printf("long int %d \n",sizeof(li));
```

```
    printf("float %d \n",sizeof(f));
```

```
    printf("double %d \n",sizeof(d));
```

```
    printf("double %d",sizeof(double));
```

```
}
```

Função que recebe a palavra reservada referente a um tipo primitivo (com ou sem modificador de tipo) ou um identificador de uma variável e retorna o número de bytes ocupado pela mesma.

Conversão de Tipos

➤ **Implícita** → dos tipos menores para os tipos maiores

Exemplos:

char → int

int → long int

float → double

Conversão de Tipos

```
#include <stdio.h>
main ()
{
    int a;
    char b;
    b='B';
    printf ("nº de bytes da variavel b = %d\n", sizeof (b));
    printf ("conteudo da variavel b = %d\n", b);
    a=b;
    printf ("nº de bytes da variavel a = %d\n", sizeof (a));
    printf ("conteudo da variavel a = %d\n", a);
}
```

Conversão de Tipos

➤ **Explícita** → de qualquer tipo para qualquer outros tipos maiores, utilizando cast **(modeladores)**.

Exemplos:

int → char

int → short int

double → float

Conversão de Tipos

```
#include <stdio.h>
main ()
{
    int a;
    char b;
    a=120;
    printf ("nº de bytes da variavel a = %d\n", sizeof (a));
    printf ("conteudo da variavel a = %d\n", a);
    b=(char)a; /*sem perda de dados*/
    printf ("nº de bytes da variavel b = %d\n", sizeof (b));
    printf ("conteudo da variavel b = %d\n", b);
}
```

Conversão de Tipos

```
#include <stdio.h>
main ()
{
    int a;
    char b;
    a=365;
    printf ("nº de bytes da variavel a = %d\n", sizeof (a));
    printf ("conteudo da variavel a = %d\n", a);
    b=(char)a; /*com perda de dados*/
    printf ("nº de bytes da variavel b = %d\n", sizeof (b));
    printf ("conteudo da variavel b = %d\n", b);
}
```

Funções Matemáticas

Funções Matemáticas

➡ ***abs ()*** /*#include <math.h>*/

➡ Calcula o valor absoluto de um inteiro.

➡ Sintaxe:

```
int abs (int n);
```

Exemplo:

```
int i;  
i = -8;  
printf ("%d", abs(i));  
/*escreve na tela 8*/
```

Funções Matemáticas

➡ ***fabs ()*** /*#include <math.h>*/

➡ Calcula o valor absoluto de um real.

➡ Sintaxe:

float fabs (float n);

Exemplo:

float f;

f = -3.7;

printf ("%f", fabs(f));

/*escreve na tela 3.70000*/

Funções Matemáticas

➡ ***pow ()*** /*#include <math.h>*/

➡ Potenciação x^y .

➡ Sintaxe:

double pow (double x, double y);

Exemplo:

```
printf ("%0.2lf", pow(2, 3));  
/* escreve na tela 8.00 */  
printf ("%0.2f", (float)pow(2, 3));  
/* escreve na tela 8.00 */
```

Funções Matemáticas

➤ ***sqrt ()*** /*#include <math.h>*/

➤ Raiz Quadrada.

➤ Sintaxe:

double sqrt (double x);

Exemplo: printf (“%.3lf”, sqrt(9));
 /* escreve na tela 3.000 */

Funções Matemáticas

➡ ***log ()*** /*#include <math.h>*/

➡ Logarítmo natural.

➡ Sintaxe:

double log (double x);

Exemplo: printf (“%.1f”, log(30));
 /* escreve na tela 3.4 */

Funções Matemáticas

➡ ***log10 ()*** /*#include <math.h>*/

➡ Logarítmo na base 10.

➡ Sintaxe:

double log10 (double x);

Exemplo: printf (“%.3lf”, log10(30));
 /* escreve na tela 1.477 */

Funções Matemáticas

➡ `cos ()` /*#include <math.h>*/

➡ Cosseno.

➡ Sintaxe:

`double cos (double x);` /* x em radianos */

Exemplo: `printf (“%.2lf”, cos(3.141592/4));`
 /* escreve na tela 0.71 */

Funções Matemáticas

➤ *sin ()* /*#include <math.h>*/

➤ Seno.

➤ Sintaxe:

double sin (double x); /* x em radianos */

Exemplo: printf (“%.2lf”, sin(3.141592/3));
 /* escreve na tela 0.87 */

Funções Matemáticas

➤ ***tan ()*** /*#include <math.h>*/

➤ Tangente.

➤ Sintaxe:

double tan (double x); /* x em radianos */

Exemplo: printf (“%.2lf”, tan(3.141592/5));
 /* escreve na tela 0.73 */