

Funções de Entrada e Saída Formatada

```
#include <stdio.h>
main ()
{
    printf("juros de 10%%");
}
```



juros de 10%

Exercício

Construa um programa, na linguagem de programação C, que escreva a string “juros de ”, o inteiro 10 e o caractere ‘%’ na tela, constituindo a seguinte frase:

juros de 10%

Funções de Entrada e Saída Formatada

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

```
main ()
```

```
{
```

```
    printf("%s%d%c", "juros de ", 10, '%');
```

```
}
```



Edição e Compilação

Linguagem de Programação C

Tradução

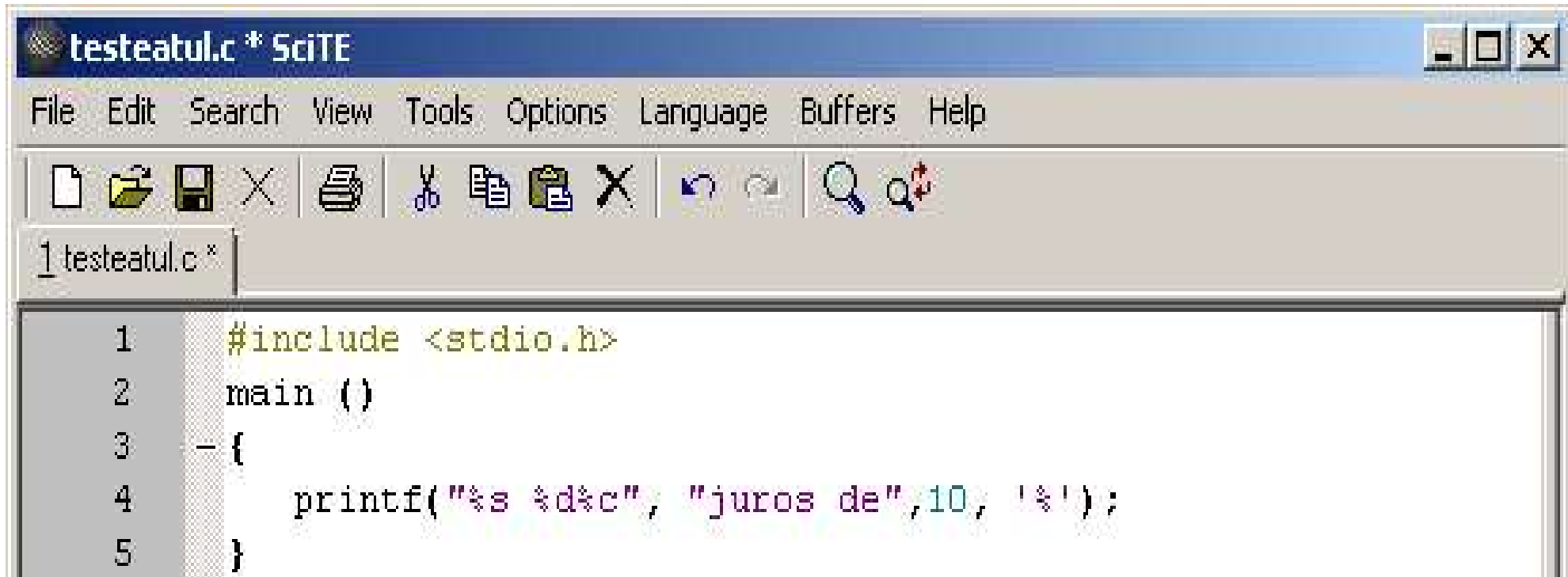
Para tornar nosso computador um ambiente de desenvolvimento, além de possuímos um gerenciador de arquivos e um editor de textos devemos instalar um pacote com um compilador (ou interpretador) e um *linkeditor* para a linguagem de programação que pretendemos utilizar.

Uma sugestão é MinGW um compilador gcc para Windows.

Existem muitos editores para código fonte. Uma grande vantagem de utilizar um é poder visualizar palavras chave, constantes e outros componentes em destaque.

Tradução

Um editor para Windows, bem leve e multi-linguagem é o SciTE.

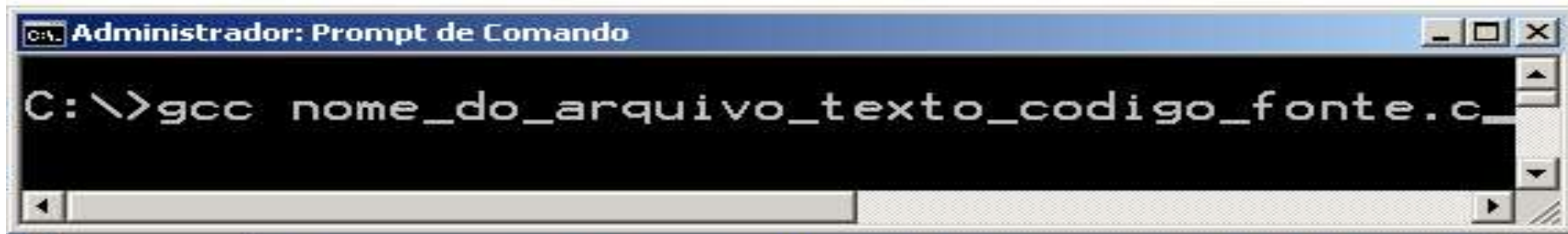


The image shows a screenshot of the SciTE text editor window. The title bar reads "testeatul.c * SciTE". The menu bar includes "File", "Edit", "Search", "View", "Tools", "Options", "Language", "Buffers", and "Help". The toolbar contains icons for file operations (new, open, save, close, print), editing (cut, copy, paste, undo, redo), and search. The main editing area shows a C program with the following code:


```
1  #include <stdio.h>
2  main ()
3  -{
4      printf("%s %d%c", "juros de", 10, '%');
5  }
```

Tradução

- Compilação no Windows
 - Após a edição do código fonte;
 - Abra um Prompt de Comando;
 - Digite:



```
C:\>gcc nome_do_arquivo_texto_codigo_fonte.c
```

- Se não ocorrer nenhum erro no processo de compilação será gerado um arquivo executável contendo o código de máquina com o nome a.exe, 

Tradução

➤ Compilação

- Para se determinar o nome do arquivo executável utilize a diretiva de compilação **-o**

➤ Exemplo:



```
C:\>gcc nome_do_arquivo_fonte.c -o nome
```

- Depois é só digitar o nome do arquivo .exe no

Prompt de Comando e pressionar a tecla *enter*

Tradução

↙ Edição e Compilação

Funcionamento no S.O. Linux...

Funções de Entrada e Saída Formatada

Parte II

Funções de Entrada e Saída Formatada

➤ printf (continuação)

É possível também indicar o tamanho do campo, alinhamento e o número de casas decimais. Para isso, utilizam-se códigos colocados entre o % e a letra que indica o tipo do formato.

Exemplos:

`%5d, %05d, %-5d`

`%10.4f, %.4f, %-10.15s`

Funções de Entrada e Saída Formatada

➤ printf (continuação)

Exercício: Construa um programa em C que utilizando-se dos códigos % escreva na saída padrão a seguinte sequência no formato apresentado:

|teste |00027| 28.37| funcionou|

Número de
colunas:

15

5

10

20

Funções de Entrada e Saída Formatada

➤ printf (continuação)

Exercício (resposta):

```
#include<stdio.h>
main ()
{
    printf("|%-15s| %05d |%10.2f| %20s |",
        "teste",27,28.37,"funcionou");
}
```

Funções de Entrada e Saída Formatada

Constantes de barra invertida

Constante	Significado
\n	new line
\"	aspas
'	apóstrofo
\0	nulo (zero decimal)
\\	barra invertida
\t	tabulação horizontal (tab)
\b	retorno do cursor

Funções de Entrada e Saída Formatada

Parte III

Funções de Entrada e Saída Formatada

➤ **scanf ()**

- função para leitura de dados;
- formato geral:

scanf (string_de_controle, lista_de_argumentos);

- *string_de_controle* → descrição de todos os valores que serão lidos, com informações de seus tipos e ordem de leitura.
- *lista_de_argumentos* → lista com os identificadores das variáveis onde os valores lidos serão armazenados, em ordem compatível com a *string_de_controle*;

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

Funções de Entrada e Saída Formatada

➡ scanf (continuação)

Observação: colocar antes de cada identificador da lista de argumentos o caractere '&'

Exemplo:

```
#include<stdio.h>
int main()
{
    char ch;
    scanf ("%c", &ch);
    ...
}
```

Funções de Entrada e Saída Formatada

Exercício:

Construa um programa, na linguagem C, que solicite ao usuário o fornecimento de um valor real, através da entrada padrão, e o retorne na saída padrão com dois dígitos de precisão.

Funções de Entrada e Saída Formatada

```
#include <stdio.h>
int main ()
{
    float valor;
    scanf("%f", &valor);
    printf("%.2f" , valor);
}
```

Funções de Entrada e Saída Formatada

```
#include <stdio.h>
int main ()
{
    float valor;
    printf("Forneca um valor real: ");
    scanf("%f", &valor);
    printf("\n0 valor fornecido foi: %.2f\n", valor);
}
```

Funções de Entrada e Saída Formatada

Exercício:

Construa um programa, na linguagem C, que receba três notas e seus respectivos pesos, através da entrada padrão, calcule a média ponderada dessas notas e exiba o resultado na saída padrão.