

---

Módulo	0782 - Programação em C/C++ - Estrutura Básica e Conceitos Fundamentais
Local	
Sessão	
Formador	
Ficha	6 - Leitura Formatada

# 1. Verdadeiro ou Falso

Assumindo que declaramos as variáveis da seguinte forma: **int x; double z; char c[10];**. Indique se as seguintes expressões estão certas ou erradas:

- a. `scanf("%d", x);`
- b. `scanf("f", &z);`
- c. `scanf("%s %s", &c);`
- d. `scanf("%c", c[9]);`
- e. `scanf("%s", c);`
- f. `scanf("d", &x);`
- g. `scanf("%d", &z);`

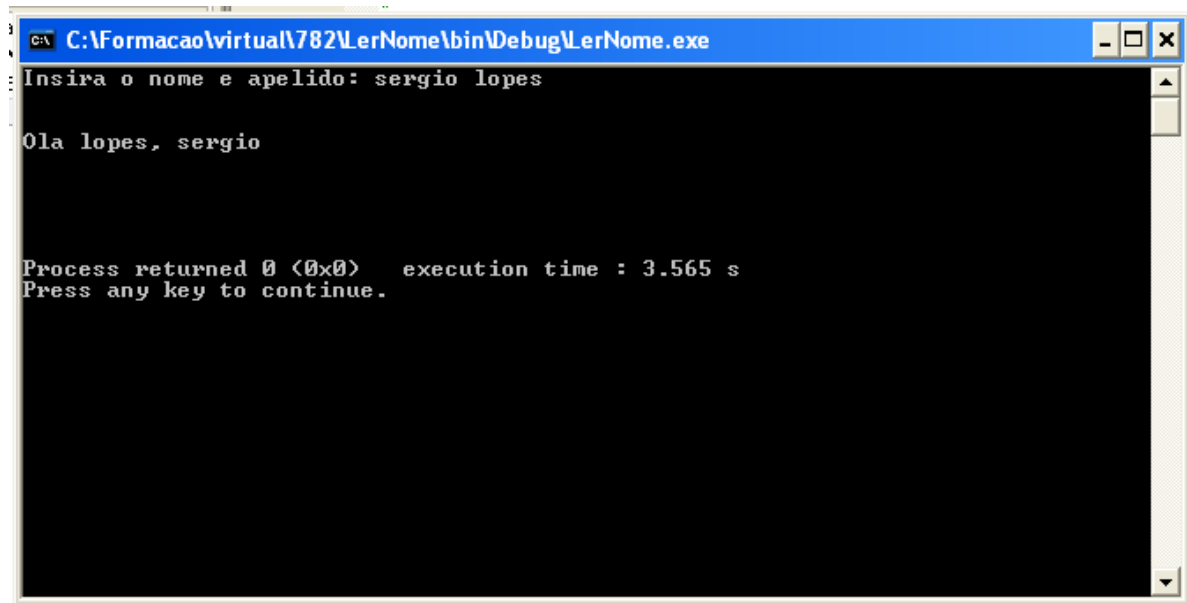
# 2. Diferença entre 'A' e "A"

Tendo em conta o que falámos sobre strings, sobre o tamanho das variáveis, e a necessidade de caracteres terminadores, explique que diferença existe entre escrever 'A' e "A", o mais completamente que conseguir.

# 3. Programar

Escreva os seguintes programas, cada um num projecto diferente:

- a) Pretende-se que seja escrito um programa que peça ao utilizador um nome e um apelido, numa única linha, e que depois de receber a informação, troque a ordem dos dados apresentando primeiro o apelido e depois o nome. Por exemplo:

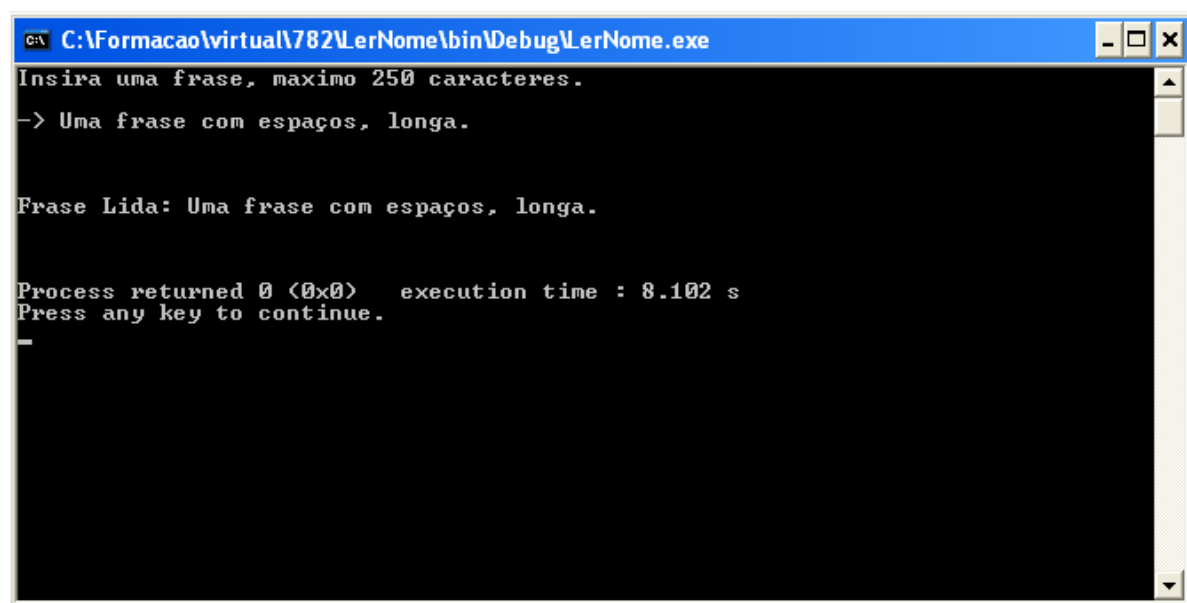


```
C:\Formacao\virtual\782\LerNome\bin\Debug\LerNome.exe
Insira o nome e apelido: sergio lopes

Ola lopes, sergio

Process returned 0 (0x0)   execution time : 3.565 s
Press any key to continue.
```

b) Pretende-se um programa que consiga ler uma frase, com 50 caracteres no máximo, mas que aceite espaços. No fim de ler a frase, o programa deve escrever a frase duas linhas abaixo. Por exemplo:



```
C:\Formacao\virtual\782\LerNome\bin\Debug\LerNome.exe
Insira uma frase, maximo 250 caracteres.
-> Uma frase com espaços, longa.

Frase Lida: Uma frase com espaços, longa.

Process returned 0 (0x0)   execution time : 8.102 s
Press any key to continue.
```

c) Crie um programa para ler a despesa mensal, para cada um dos doze meses do ano, e que no fim mostre a média de gastos. Por exemplo:

d) Pretende-se um programa que receba um nome, com um máximo de 20 caracteres, uma data de nascimento, com um máximo de 10 caracteres e no formato dd/mm/aaaa e o sexo do utilizador, guardando **Feminino** ou **Masculino**. Depois de ler estes dados deve apresentar um quadro, desenhado com caracteres seguindo o exemplo apresentado. Consulte a tabela de códigos para caracteres na secção de notas.

e) Um programa que leia uma palavra mas que pare de ler assim que encontrar uma vogal. Por exemplo, se for introduzida a frase **Raposa azul**, o resultado apresentado deve ser apenas **R**, já que a leitura deve parar na primeira vogal, o primeiro **a** da palavra **Raposa**.

f) O programa deve permitir ler um código composto por uma letra seguida de um número, por exemplo **R78**. Depois de lido o código, o programa deve apresentar o mesmo código mas colocar zeros na parte dos números de forma a que o código tenha sempre 8 dígitos. Pegando no exemplo anterior, o programa deveria apresentar o resultado **R00000078**.

g) Escreva um programa que leia um número em hexadecimal e o mostre em decimal, por exemplo **FF** equivale a 255.

---

h) Escreva um programa que leia uma data completa, separando cada um dos componentes. Se a data introduzida for **5 de Outubro de 2010, 18:15**, o programa deve guardar em variáveis separadas o dia (5), o mês (Outubro) o ano (2010) a hora (18) e os minutos (15). Deve depois mostrar essa informação na usando a frase **Nascido no dia 5 do mês de Outubro do ano de 2010, às 18 horas e 15 minutos**.

i. Escreva um programa que receba o número de semanas que faltam para o Natal, e que mostre ao utilizador o número de segundos que faltam até ao Natal.

## 4. Implementar

Implemente num projecto novo, a resposta a cada uma das seguintes perguntas:

1. Como imprimia o endereço de memória de uma variável?
2. Como imprimia o valor de uma variável?
3. Como lia o um valor decimal, com vírgulas e 4 casas decimais, e imprimia apenas com uma casa decimal?

## Notas

### Uso de &

Se a variável for declarada como vector, e vai ser usada para guardar texto, não precisa de **&**. Caso contrário, é necessário colocar o carácter **&** para indicar o endereço de memória para onde pretendemos escrever.

### Tabela com formatações do printf

%-	Alinha o texto à esquerda
%05	Preenche com 5 zeros o texto, se o texto tiver menos de 5 caracteres, se tiver 5 ou mais não faz alterações
%5	Igual à formatação anterior mas usa espaços em vez de zeros
%.3	Formata um número decimal com 3 casas
%c	Um carácter (exemplo: <i>a</i> )
%i ou %d	Um número inteiro em decimal (exemplo: 1)
%f	Um valor com casas decimais
%p	Um ponteiro
%s	Uma string

### Tabela com formatações do scanf

%c	Um carácter (exemplo: <i>a</i> )
%i ou %d	Um número inteiro em decimal (exemplo: 1)

%e ou %f ou %g	Um float (exemplo: 5.9785)
%o	Um número em octal
%s	Uma simples string
%x	Um número em hexadecimal
%p	Um ponteiro
%n	Um inteiro equivalente ao número de caracteres escritos até ali naquela iteração
%u	Um inteiro sem sinal
%[]	Um conjunto de caracteres
%%	Símbolo %
^	Nega o conjunto seguinte, por exemplo ler tudo excepto A, B e C: scanf("%^[ABC]", &var);

## Códigos de caracteres

128	Ç	144	É	160	á	176	☐	192	Ł	208	⌌	224	α	240	≡
129	ü	145	æ	161	í	177	☐	193	Ł	209	Ŧ	225	β	241	±
130	é	146	Æ	162	ó	178	☐	194	Ŧ	210	Π	226	Γ	242	≥
131	â	147	ô	163	ú	179		195	†	211	⌌	227	π	243	≤
132	ä	148	ö	164	ñ	180	†	196	—	212	⌌	228	Σ	244	∫
133	à	149	ò	165	Ñ	181	†	197	†	213	ƒ	229	σ	245	∫
134	â	150	û	166	²	182		198	†	214	π	230	μ	246	÷
135	ç	151	ù	167	°	183	π	199		215	†	231	τ	247	≈
136	ê	152	ÿ	168	¿	184	¶	200	⌌	216	†	232	Φ	248	°
137	ë	153	Ö	169	┐	185	¶	201	ƒ	217	∫	233	⊕	249	.
138	è	154	Ü	170	┐	186		202	⌌	218	┐	234	Ω	250	.
139	ï	155	◊	171	½	187	¶	203	Ŧ	219	■	235	δ	251	√
140	î	156	£	172	¼	188	¶	204	†	220	■	236	∞	252	Δ
141	ì	157	¥	173	¡	189	¶	205	=	221	■	237	φ	253	²
142	Ä	158	£	174	«	190	¶	206	†	222	■	238	ε	254	■
143	Å	159	ƒ	175	»	191	¶	207	⌌	223	■	239	∩	255	

Source : [www.LookupTables.com](http://www.LookupTables.com)