

Curso Profissional: Programador/a de Informática
PSD – 10.º ano: UFCD 0809 - Programação em C/C++ - fundamentos

Ficha de Trabalho 2

Ano letivo 21/22

Programação básica em linguagem C

- Em C, um programa começa com a função **main()**
- O código a executar é colocado entre { }
- Um bloco é formado por qualquer conjunto de instruções entre { }
- Cada instrução deve ser seguida de ponto e vírgula
- A disposição do código é arbitrária e depende das preferências de cada programador
- O C faz distinção entre maiúsculas e minúsculas - diz-se **case sensitive**
- As strings em C são delimitadas pelo carácter aspas “, os caracteres pelo carácter plica ‘
- A função **printf ()** não faz parte da linguagem C. Pertence à sua extensa biblioteca de funções e usa-se para escrever mensagens no ecrã
- A diretiva **#include** permite indicar ao compilador que bibliotecas se querem usar
- Para termos acesso à função referida no ponto anterior e a outras funções de I/O devemos incluir nos programas o ficheiro **stdio.h**, através da diretiva do pré-processor **#include <stdio.h>**
- As linhas começadas por # (# include, #define, etc ...) não são C, mas sim diretivas do pré-processor, por isso não devem ser seguidas de ;
- A execução do programa é feita através da linha de comando, invocando o executável resultante da compilação
- A representação de caracteres especiais ou de caracteres que de outra forma seriam difíceis de representar, faz-se através de um conjunto de 2 ou + caracteres, sendo em geral o 1º carácter a barra invertida (\)
- O símbolo \ é utilizado para retirar o significado especial que um carácter tem.

A lista de caracteres que podem ter que se representam, precedidos do carácter especial `\`, é:

Carácter	Sequência de Escape	Valor ASCII
Campainha	<code>\a</code> ou <code>\7</code>	007
Retrocesso	<code>\b</code>	008
Tabulação horizontal	<code>\t</code>	009
Tabulação vertical	<code>\v</code>	011
Nova linha	<code>\n</code>	010
Aspas	<code>\"</code>	034
Apóstrofe	<code>\'</code>	039
Interrogação	<code>\?</code>	063
Barra invertida	<code>\\</code>	092
Nulo	<code>\0</code>	000
Carácter %	<code>%%</code>	037
Carriage Return(CR)	<code>\r</code>	013

- Os comentários são escritos entre `/*` e `*/` ou usando `//`, se se tratar de uma única linha, e são ignorados pelo compilador (a menos que se encontrem dentro de uma string);
- Não podem existir comentários dentro de comentários.

Carateres gráficos (imprimíveis) ver fim da ficha

Exercícios:

1. Qual a função que deve estar presente em todos os programas em C ?
2. Como devem terminar todas as instruções em C ?
3. Como é delimitado um bloco em C ?
4. A função **printf** faz parte integrante da linguagem C ?
5. Para que serve a linha **#include <stdio.h>** num programa?
6. A extensão **.h** indica que o ficheiro correspondente é composto por ...?
7. Os ficheiros com extensão **.h** são também conhecidos por ... ?
8. Dentro de uma string podemos usar maiúsculas? Justifica.

9. prog0101.c

```
main()  
{  
}
```

Porque razão não se utilizou a linha **#include <stdio.h>** no programa prog0101.c ?

10. Qual o significado de **stdio**?

11. Identifica os erros de compilação que seriam detetados nos seguintes programas:

11.1

```
/*  
 * Copyright: Asneira Suprema de Software!!!  
 */  
  
#include <stdio.h>  
Main()  
{  
    printf("Hello World");  
}
```

11.2

```
/*  
 * Copyright: Asneira Suprema de Software!!!  
 */  
  
#include <stdio.h>  
main  
{  
    printf("Hello World");  
}
```

11.3

```
/*  
 * Copyright: Asneira Suprema de Software!!!  
 */  
  
#include <stdio.h>  
main()  
{  
    print("Hello World");  
}
```

11.4

```
/*
 * Copyright: Asneira Suprema de Software!!!
 */

#include <stdio.h>
main()
{
    printf("Hello") (" World");
}
```

11.5

```
/*
 * Copyright: Asneira Suprema Software!!!
 */

#include <stdio.h>
main()
{
    printf("Hello World");
}
```

11.6

```
/*
 * Copyright: Asneira Suprema de Software!!! */
*/

#include <stdio.h>
main()
{
    printf(Hello World);
}
```

12. Escreve um programa em C que apresente o seguinte *output*:

- 1- Clientes
- 2- Fornecedores
- 3- Faturas
- 0- Sair

13. Escreve um programa em C que apresente duas linhas com a *string* "Aqui vai um Apito", ouvindo-se no final de cada *string* um sinal sonoro.

14. Os comentários devem ser escritos (assinala a resposta correta):

- a) Antes de qualquer instrução do programa.
- b) Depois de todas as instruções.
- c) Antes do main.
- d) Sempre que o programador ache necessário ou conveniente.

15. Um programa em C, que tenha comentários no seu código, é em relação a outros que não tenha (assinale a resposta correta):

- a) Mais rápido a executar.
- b) Mais lento a executar.
- c) Executado praticamente à mesma velocidade pois os comentários requerem uma utilização ínfima do CPU.
- d) Executados à mesma velocidade, pois os comentários são simplesmente ignorados pelo compilador, não havendo qualquer reflexo deles em tempo de execução.

16. Escreve um programa que coloque no ecrã a seguinte frase:

Bem-vindos ao /Mundo\ da programação em “C”

17. Escreve um programa que coloque no ecrã uma árvore com o seguinte formato:

*

 / | \

18. Escreve um programa que coloque no ecrã o seguinte *output*

Total	=	100%
IVA	=	17%
IRS	=	15%
<hr/>		
Liq.	=	68%

19. Experimenta a função `puts` (“Hello World”); (*put string*) para escrever a *string* “Hello World” e indica qual a diferença entre esta e a função *printf*. (Nota: esta função também faz parte do *stdio.h*).

20. Escreve um programa em C que indique qual o significado dos seguintes caracteres especiais - \n , \\ , \t , %%

Com o seguinte output:

\n	-	<ENTER>
\\	-	\
\t	-	<TAB>
%%	-	%

Carateres gráficos (imprimíveis)

Bin	Oct	Dec	Hex	Sinal	Bin	Oct	Dec	Hex	Sinal	Bin	Oct	Dec	Hex	Sinal
0010 0000	040	32	20	(espaco)	0100 0000	100	64	40	@	0110 0000	140	96	60	`
0010 0001	041	33	21	!	0100 0001	101	65	41	A	0110 0001	141	97	61	a
0010 0010	042	34	22	"	0100 0010	102	66	42	B	0110 0010	142	98	62	b
0010 0011	043	35	23	#	0100 0011	103	67	43	C	0110 0011	143	99	63	c
0010 0100	044	36	24	\$	0100 0100	104	68	44	D	0110 0100	144	100	64	d
0010 0101	045	37	25	%	0100 0101	105	69	45	E	0110 0101	145	101	65	e
0010 0110	046	38	26	&	0100 0110	106	70	46	F	0110 0110	146	102	66	f
0010 0111	047	39	27	'	0100 0111	107	71	47	G	0110 0111	147	103	67	g
0010 1000	050	40	28	(0100 1000	110	72	48	H	0110 1000	150	104	68	h
0010 1001	051	41	29)	0100 1001	111	73	49	I	0110 1001	151	105	69	i
0010 1010	052	42	2A	*	0100 1010	112	74	4A	J	0110 1010	152	106	6A	j
0010 1011	053	43	2B	+	0100 1011	113	75	4B	K	0110 1011	153	107	6B	k
0010 1100	054	44	2C	,	0100 1100	114	76	4C	L	0110 1100	154	108	6C	l
0010 1101	055	45	2D	-	0100 1101	115	77	4D	M	0110 1101	155	109	6D	m
0010 1110	056	46	2E	.	0100 1110	116	78	4E	N	0110 1110	156	110	6E	n
0010 1111	057	47	2F	/	0100 1111	117	79	4F	O	0110 1111	157	111	6F	o
0011 0000	060	48	30	0	0101 0000	120	80	50	P	0111 0000	160	112	70	p
0011 0001	061	49	31	1	0100 0000	100	64	40	@	0111 0001	161	113	71	q
0011 0010	062	50	32	2	0100 0001	101	65	41	A	0111 0010	162	114	72	r
0011 0011	063	51	33	3	0100 0010	102	66	42	B	0111 0011	163	115	73	s
0011 0100	064	52	34	4	0100 0011	103	67	43	C	0111 0100	164	116	74	t
0011 0101	065	53	35	5	0100 0100	104	68	44	D	0111 0101	165	117	75	u
0011 0110	066	54	36	6	0100 0101	105	69	45	E	0111 0110	166	118	76	v
0011 0111	067	55	37	7	0100 0110	106	70	46	F	0111 0111	167	119	77	w
0011 1000	070	56	38	8	0100 0111	107	71	47	G	0111 1000	170	120	78	x
0011 1001	071	57	39	9	0100 1000	110	72	48	H	0111 1001	171	121	79	y
0011 1010	072	58	3A	:	0100 1001	111	73	49	I	0111 1010	172	122	7A	z
0011 1011	073	59	3B	;	0100 1010	112	74	4A	J	0111 1011	173	123	7B	{
0011 1100	074	60	3C	<	0100 1011	113	75	4B	K	0111 1100	174	124	7C	
0011 1101	075	61	3D	=	0100 1100	114	76	4C	L	0111 1101	175	125	7D	}
0011 1110	076	62	3E	>	0100 1101	115	77	4D	M	0111 1110	176	126	7E	~
0011 1111	077	63	3F	?	0100 1110	116	78	4E	N					