

Ficha de Avaliação Sumativa

EFA Tecnológico - Técnico de Informática e Sistemas

UFCD:0785 – Programação em C – C++ - Formas complexas

Duração: 120 minutos

Nome Formando: Trigo Pato

Data: 31/08/12

Formador: João Leitão

Classificação: Bom 16,7 pontos

Notas Gerais:

Esta prova de natureza teórica divide-se em três grupos e tem a duração de 120 minutos sem tolerância.

Utilize apenas caneta ou esferográfica de tinta indelével, azul ou preta.

Não é permitido o uso de corrector. Em caso de engano, deve riscar, de forma inequívoca, aquilo que pretende que não seja classificado.

Não é permitido o uso de auxiliares.

As respostas ilegíveis ou que não possam ser identificadas são classificadas com zero pontos.

Rubrique no canto superior direito todas as folhas da prova.

Qualquer tentativa de cópia será punida com a anulação da prova.

As cotações dos itens encontram-se descritas no quadro seguinte:

Grupo	Questão	Alínea	Cotação (Pontos)	TOTAL
Grupo I	1.		0,5	2
	2.		0,5	
	3.		0,5	
	4.		0,5	
Grupo II	1.		x 3	6
	2.		x 3	
Grupo III	1.		2	12
	2.		2	
	3.		x 3	
	4.		x 3	
	5.		2	
				20

Trigo Preto

Grupo I

Atenção! Se apresentar mais de uma resposta, ou resposta ambígua, a questão será anulada.

1. Indique, para cada afirmação, se é verdadeira ou falsa, justificando no caso de ser falsa:

- ☒ a) Na declaração de um vector é obrigatório colocar o conteúdo de cada posição.

~~Não~~ Não, na declaração, Não é Obrigatório por o conteúdo de um vector

- b) Num vector com N elementos, a primeira posição é N e a última é 0.

Não, num vector com N elementos a primeira é 0 e a última é N

- ☒ c) É possível alterar a indicação de qual a primeira posição de um vector. Por exemplo, é possível dizer que a primeira posição de um vector passa a ser a posição 5.

Não é possível ~~alterar~~ a primeira posição e ~~temper a que se de um vector~~ posteriormente e que se vai buscar a posição

- d) É possível utilizar o conteúdo de uma posição de um vector na condição da estrutura de decisão if.

Sim ~~temper~~ o conteúdo de uma posição de um vector e utilizado no "IF"

Fig. 1

Grupo II

1. Indique qual o resultado final para o utilizador após a execução do seguinte fragmento de código:

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

```
int main(){
```

```
int xpto[5];
```

```
int i;
```

```
int res;
```

```
xpto[0]=1;
```

```
xpto[1]=0;
```

```
xpto[2]=0;
```

```
xpto[3]=1;
```

```
xpto[4]=0;
```

```
res=xpto[0];
```

```
for(i=1;i<5;i=i+2){
```

```
    xpto[i]=xpto[i]+res;
```

```
}
```

```
res=1;
```

```
for(i=0;i<5;i=i+1){
```

```
    res=res+xpto[i];
```

```
}
```

```
res=res*i;
```

```
if(res>0){
```

```
    res=res-i;
```

```
}
```

```
else{
```

```
    res=res+1;
```

```
}
```

```
printf("%d", res);
```

```
for(i=0;i<5;i=i+1){
```

```
    printf("%d",xpto[i]);
```

```
}
```

```
return(0);
```

```
}
```

$xpto[0]=1$

$xpto[1]=0$

$xpto[2]=0$

$xpto[3]=1$

$xpto[4]=0$

$res = xpto[0]$

$res = 1$

$xpto[1] = 0 + 1 = 1$

$xpto[2] = 0 + 1 = 1$

$xpto[3] = 1 + 1 = 2$

$xpto[4] = 0 + 2 = 2$

$res = 1 + 1 = 2$

$res = 2 + 1 = 3$

$res = 3 + 0 = 3$

$res = 3 + 1 = 4$

$res = 4 + 2 = 6$

$res = 6 + 2 = 8$

$res = 8 + 2 = 10$

$res = 10 + 2 = 12$

$res = 12 + 2 = 14$

$res = 14 + 2 = 16$

$res = 16 + 2 = 18$

$res = 18 + 2 = 20$

$res = 20 + 2 = 22$

$res = 22 + 2 = 24$

$res = 24 + 2 = 26$

$res = 26 + 2 = 28$

$res = 28 + 2 = 30$

$res = 30 + 2 = 32$

$res = 32 + 2 = 34$

$res = 34 + 2 = 36$

$res = 36 + 2 = 38$

$res = 38 + 2 = 40$

$res = 40 + 2 = 42$

$res = 42 + 2 = 44$

$res = 44 + 2 = 46$

$res = 46 + 2 = 48$

$res = 48 + 2 = 50$

$res = 50 + 2 = 52$

$res = 52 + 2 = 54$

$res = 54 + 2 = 56$

$res = 56 + 2 = 58$

$res = 58 + 2 = 60$

$res = 60 + 2 = 62$

$res = 62 + 2 = 64$

$res = 64 + 2 = 66$

	1	2	3	4	5
1	0	0	1	0	1
2	1	1	2	2	3
3	2	2	3	3	4
4	3	3	4	4	5
5	4	4	5	5	6

5
4
3
2
1

25
20

res = 20

Obrigatório mostrar cálculos intermédios!

v_1 $printf("%d", 1)$; v_2 $(printf("%d", 0))$ v_3 $(printf("%d", 2))$ v_4 $(printf("%d", 0))$

2. Indique qual o resultado final para o utilizador após a execução do seguinte fragmento de código:

```
#include<stdio.h>
int main(){

    int vector[4];
    int i;
    int res;
    int w;
    int init;

    init=1;
    res=2;

    for(i=0;i<4;i=i+1){
        vector[i]=init;
    }

    init=init+1;
    w=0;
    res=res+1;

    for(i=0;i<4;i=i+3){
        res=res+vector[i];
    }

    res=res+w;
    init=init-1;

    for(i=0;i<4;i=i+1){
        vector[i]=vector[i]+i;
    }

    for(i=3;i>=0;i=i-1){
        w=i+vector[i]-init;
    }

    res=res+i;
    w=w-i;

    printf("%d\n", res);
    printf("%d\n", w);

    for(i=0;i<4;i=i+1){
        printf("%d\n",vector[i]);
    }

    return(0);
}
```

Trigo

Handwritten calculations and notes:

- 5 (written in red)
- -2 (written in blue)
- $VE01 = 1$
- $VE13 = 2$
- $VE21 = 3$
- $VE32 = 4$

Obrigatório mostrar cálculos intermédios!

~~Ans~~

	0	1	2	3	4
1	1	1	1	1	
1	2	3	4		
0	2	4	6		
	6	7	8		

rs	2	3	4	5	5	9
init	1	2	1			
w	0	6	2			

for (i=0; i<4; i=i+1) {

x Vector[0] = 1

Vector[1] = 1

Vector[2] = 1

x Vector[3] = 1

~~Vector[4] = 1~~

init = 1 + 1 = 2

w = 0

rs = 2 + 1 = 5

for (i=0; i<4; i=i+1) {

v0 rs = 3 + 1 = 4

v3 rs = 4 + 1 = 5

rs = 5 + 0 = 5

init = 2 - 1 = 1

for (i=0; i<4; i=i+1) {

v0 = 1 + 0 = 1

v1 = 1 + 1 = 2

v2 = 1 + 2 = 3

v3 = 1 + 3 = 4

for (i=3; i>0; i=i-1)

w = v3 = 3 + 4 - 1 = 6

v2 = 2 + 3 - 1 = 4

v1 = 1 + 2 - 1 = 2

v0 = 0 + 1 - 1 = 0

~~rs = 5 + 4 = 9~~

~~v2 rs = 5 + 2 = 7~~

~~v1 rs = 4 + 1 = 5~~

~~rs rs = 5 + 0 = 5~~

~~v3 w = 6 - 4 = 2~~

~~v2 w = 2 - 2 = 0~~

~~v1 w = 0 - 1 = -1~~

~~v0 w = -1 - 0 = -1~~

printf("%d\n", rs)

printf("%d\n", w)

for (i=0; i<4; i=i+1)

Grupo III (Utilização de vectores)

1. Utilizando vectores, crie um programa que receba 5 números inteiros positivos e verifique para cada posição se esse número é superior ou igual a 20. Caso seja, deverá receber uma mensagem a dizer: "Número superior ou igual a 20". Caso contrário, receberá "Número inferior a 20".

2. Utilizando vectores, crie um programa que receba 4 números do utilizador e consoante o valor guardado no vector, realize as seguintes acções:
Se o valor guardado for superior a 1, então cria uma pasta com o nome pasta1, se for igual a 2, cria uma pasta com o nome pasta2, se for igual a 3 abre o site www.google.pt. Caso contrário avisa o utilizador que o número não tem acção prevista.

*Introdução
Ciclo for*

3. Utilizando vectores, crie um programa que receba do utilizador 5 alturas, em centímetros, e que mostre ao utilizador qual a média e desvio padrão sendo que:

$$\text{Média} = SA / 5$$

$$\text{Variância} = SA - \text{Média}$$

SA corresponde à soma de todos os elementos do vector.

*Calculando para
o ciclo for!*

4. Utilizando vectores, crie um programa que receba para cada produto, um preço de compra e um preço de venda. Pretende-se receber os preços para 5 produtos e o objectivo é que os valores fiquem em posições adjacentes (lado a lado). Ou seja, para o produto 1, o preço de compra fica na posição 0 e o preço de venda na posição 1 e assim sucessivamente para os restantes produtos. Pretende-se mostrar ao utilizador, para cada produto, qual o lucro obtido através da diferença entre o preço de venda e o preço de compra.

Compasso!

5. Utilizando vectores, crie um programa que dados 5 números devolva a diferença entre a quantidade de números positivos e negativos, acompanhada de uma mensagem explicativa do sucedido, isto é, se essa diferença for superior a 0 deverá receber a mensagem: "Existem mais números positivos que negativos". Se for inferior a 0: "Existem mais números negativos que positivos". Se for igual a 0: "O número de positivos e negativos é igual".

*Calculando
diferença
X = X + 1*

Exemplo:

1	2	3	-2	-1
---	---	---	----	----

Resultado para o utilizador:

1

Existem mais números positivos que negativos.