

Revisão da linguagem C

01. A função **main()** deve existir em alguma parte de um programa em C e marca o ponto de início da execução.

<input checked="" type="checkbox"/> A.	Verdadeiro
<input type="checkbox"/> B.	Falso

02. Em um programa em C, os nomes **num** e **Num** podem ser usados indistintamente em diferentes partes do programa para referenciar a mesma variável.

<input type="checkbox"/> A.	Verdadeiro
<input checked="" type="checkbox"/> B.	Falso

03. A opção que inclui apenas nomes válidos para variáveis na linguagem C é:

<input type="checkbox"/> A.	lf, a_b_2, H789, _yes
<input type="checkbox"/> B.	i, j, int, obs
<input type="checkbox"/> C.	9xy, a36, x*y, --j
<input type="checkbox"/> D.	2_ou_1, \fim, *h, j
<input checked="" type="checkbox"/> E.	Nenhuma das opções anteriores

Sobre o trecho de programa abaixo

```
... int main()
{
    char opcao;
    int i = 1;
    opcao = 'B';
    ...
}
```

pode-se afirmar que é:

<input type="checkbox"/> A.	Válido na linguagem C
<input checked="" type="checkbox"/> B.	Não válido na linguagem C

04. Em C, "**v**" e '**v**' representam a mesma constante.

<input type="checkbox"/> A.	Verdadeiro
<input checked="" type="checkbox"/> B.	Falso

05. O programa

```
#include <stdio.h>
main()
{
    int numero;
    scanf("%d",&numero);
    printf("%d",numero);
}
```

Lê uma variável pelo teclado e a imprime na tela.

<input checked="" type="checkbox"/> A.	Verdadeiro
<input type="checkbox"/> B.	Falso

06. A instrução **#include <stdio.h>** no programa anterior é colocada para que se possa utilizar funções tais como **scanf** e **printf**.

<input checked="" type="checkbox"/> A.	Verdadeiro
<input type="checkbox"/> B.	Falso

07. Na linguagem C, cada comentário deve ser restrito a uma única linha de código.

<input type="checkbox"/> A.	Verdadeiro
<input checked="" type="checkbox"/> B.	Falso

08. O programa a seguir está correto.

```
int main()
{
    int x=3; y=5, z=7;
    printf("Os números são: %d %d %d\n,x,y,z,w)
}
```

<input type="checkbox"/> A.	Verdadeiro
<input checked="" type="checkbox"/> B.	Falso (qual o erro?) R: Variável w não declarada.

09. Textos delimitados por **/*** (início) e ***/** (término) são ignorados pelo computador na linguagem C.

<input type="checkbox"/> A.	Verdadeiro
<input checked="" type="checkbox"/> B.	Falso

10. O que faz o seguinte programa em C?

```
#include <stdio.h>
main()
{
    int vlr =6;
    printf ("\n Valor = %d ", vlr);
}
```

<input type="checkbox"/> A.	Nada
<input checked="" type="checkbox"/> B.	Imprime: Valor = 6
<input type="checkbox"/> C.	Imprime: \n O valor de vlr = %d
<input type="checkbox"/> D.	Pula para a próxima linha e imprime: Valor = 6
<input type="checkbox"/> E.	Nenhuma das alternativas anteriores está correta.

11. Programas codificados em C devem conter pelo menos a função *main()*.

<input checked="" type="checkbox"/> A.	Verdadeiro
<input type="checkbox"/> B.	Falso

12. Em C, pares de chaves (**{ }**) servem sempre de delimitadores para blocos de código.

<input checked="" type="checkbox"/> A.	Verdadeiro
<input type="checkbox"/> B.	Falso

13. Em C, uma linha inteira de código equivale a um **comando**, devendo ser encerrada com um ponto-e-vírgula (;).

<input checked="" type="checkbox"/> A.	Verdadeiro
<input type="checkbox"/> B.	Falso

14. Qual a saída produzida pelo trecho de código a seguir:

```
int x;
for (x = 35 ; x > 0 ; x=x/3)
    printf("%d ", x);
```

<input checked="" type="checkbox"/> A.	35 11 3 1
<input type="checkbox"/> B.	11 3 1
<input type="checkbox"/> C.	11 3 1 0
<input type="checkbox"/> D.	35 11 3
<input type="checkbox"/> E.	Nenhuma das opções anteriores

15. Caso o nome da função seja escrito incorretamente em um programa em C, o *linker* indicará para o programador o erro de digitação e lista o conteúdo da biblioteca na qual a função se encontra, a fim de que o programador digite corretamente o nome da função.

<input checked="" type="checkbox"/> A.	Verdadeiro
<input type="checkbox"/> B.	Falso

16. O trecho de código abaixo

```
#include <stdio.h>
int main()
{
    int i1;
    printf("Entre com o primeiro valor:");
    scanf( "%d", &i1 );
    printf( "O valor digitado foi %d\n", i1 );
    return 0;
}
```

<input type="checkbox"/> A.	Imprimirá na tela uma mensagem para a entrada de um valor e receberá o valor do teclado, imprimindo-o na tela sem mais nada.
<input type="checkbox"/> B.	Imprimirá na tela uma mensagem para a entrada de um valor e receberá o valor do teclado, imprimindo a mensagem "O valor digitado foi" seguido do valor digitado, por sua vez seguido do símbolo %.
<input checked="" type="checkbox"/> C.	Imprimirá na tela uma mensagem para a entrada de um valor e receberá o valor do teclado, imprimindo a mensagem "O valor digitado foi" seguido do valor digitado.
<input type="checkbox"/> D.	Imprimirá na tela uma mensagem para a entrada de um valor e, em seguida, será encerrado.
<input type="checkbox"/> E.	Nenhuma das opções anteriores

17. Se um comando executável referenciar uma variável que não foi anteriormente declarada, será produzido um erro de sintaxe:

<input checked="" type="checkbox"/> A.	Verdadeiro
<input type="checkbox"/> B.	Falso

18. O seguinte trecho de código

```
int x,y;
int a = 14, b = 3;
x = a/b;
y = a%b;
z = x/y;
```

gerará como resultados:

A.	x = 4.66666, y = 2 e z = 2
B.	x = 4, y = 0.66666 e z = 2
<input checked="" type="checkbox"/>	x = 4, y = 2 e z = 2
D.	x = 4.66666, y = 0.66666 e z = 2
E.	Nenhuma das alternativas anteriores

19. Seja o seguinte trecho de programa:

```
int i=3, j=5;

int *p, *q;

p = &i;

q = &j;
```

Qual é o valor das seguintes expressões ?

a) $p == \&i$; b) $*p - *q$; c) $**\&p$ d) $3* - *p / (*q) + 7$

T -2 3 6

20. Qual é o resultado do seguinte programa? Suponha o endereço de memória inicial de vet 4000

```
#include <conio.h>
```

```
int main(){

    float vet[5] = { 1.1,2.2,3.3,4.4,5.5 };

    float *f;

    int i;

    f = vet;

    printf("contador/valor/valor/endereco/endereco");

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

        printf("\ni = %d",i);

        printf(" vet[%d] = %.1f",i, vet[i]);

        printf(" *(f + %d) = %.1f",i, *(f+i));

        printf(" &vet[%d] = %X",i, &vet[i]);

        printf(" (f + %d) = %X",i, f+i);

    }

    return 0; }
```

Saída:

i=0	vet[0]=1.1	*(f+0)=1.1	&vet[0]=4000	(f+0)=4000
i=1	vet[1]=2.2	*(f+1)=2.2	&vet[1]=4004	(f+1)=4004
i=2	vet[2]=3.3	*(f+2)=3.3	&vet[2]=4008	(f+2)=4008
i=3	vet[3]=4.4	*(f+3)=4.4	&vet[3]=4012	(f+3)=4012
i=4	vet[4]=5.5	*(f+4)=5.5	&vet[4]=4016	(f+4)=4016

21. Assumindo que **pulo[]** é um vetor do tipo **int**, quais das seguintes expressões referenciam o valor do terceiro elemento da matriz?

- X) ***(pulo + 2)** b) ***(pulo + 4)** c) **pulo + 4** X) **pulo + 2**

22. O que fazem os seguintes programas?

<pre>#include <conio.h> #include <stdio.h> int main(){ int vet[] = {4,9,13}; int i; for(i=0;i<3;i++){ printf("%d ",*(vet+i)); } return 0; }</pre>	<pre>#include <conio.h> #include <stdio.h> int main(){ int vet[] = {4,9,13}; int i; for(i=0;i<3;i++){ printf("%p ",vet+i); } return 0; }</pre>
--	---

O primeiro mostra os elementos do vetor (4,9,13) e o segundo os endereços de memória destes elementos no vetor.

23. Seja **vet** um vetor de 4 elementos: **TIPO vet[4]**. Supor que depois da declaração, **vet** esteja armazenado no endereço de memória 4092 (ou seja, o endereço de **vet[0]**). Supor também que na máquina usada uma variável do tipo **char** ocupa 1 byte, do tipo **int** ocupa 2 bytes, do tipo **float** ocupa 4 bytes e do tipo **double** ocupa 8 bytes.

Qual o valor de **vet+1**, **vet+2** e **vet+3** se:

- a) **vet** for declarado como **char**? 4093, 4094, 4095
- b) **vet** for declarado como **int**? 4094, 4096, 4098
- c) **vet** for declarado como **float**? 4096, 4100, 4104
- d) **vet** for declarado como **double**? 4100, 4108, 4116

24. Faça uma função que recebe por parâmetro o raio de uma esfera e calcula o seu volume ($v = \frac{4}{3} \cdot P \cdot R^3$)

float volume(float r)

```
{
    return 4/3*3.1415*r*r*r;
}
```

25. Faça uma função que recebe por parâmetro um valor inteiro e positivo e retorna o valor 'V' caso o valor seja primo e 'F' em caso contrário.

A resposta é só a primeira função, mas quem quiser rodar, segue o programa completo.

```
char eprimo(int n)
```

```
{
    int i,d=0;
    for(i=2;i<n;i++)
        if(n%i==0)
            d++;
    if(n>1&& d==0)
        return 'V';
    else
        return 'F';
}
```

```
int main()
```

```
{
    int valor;
    char primo;
    printf("\nDigite um valor inteiro: ");
    scanf("%d",&valor);
    primo=eprimo(valor);
    if(primo=='V')
        printf("\nO numero e primo.");
    else
        printf("\nO numero nao e primo.");
    return 0;
}
```