

Linguagem de programação I

Exercícios

- 1) Seja a seguinte sequência de instruções em um programa em C:

```
int *pti;  
int i=10;  
pti = &i;
```

Qual afirmativa é falsa?

- a) pti armazena o endereço de i
- b) *pti é igual a 10
- c) ao se executar *pti=20, i passará a ter o valor 20
- d) pti é igual a 10

- 2) Se i e j, **mat[5]={1,2,3,4,5}** são variáveis inteiras e **p** e **q** são ponteiros para inteiros, quais das seguintes expressões de atribuição são ilegais?

- | | |
|---------------|----------------------|
| a) p = mat; | b) q = &p; |
| c) q=&j; | d) *q = &j; |
| d) i = (*&)j; | e) i = (++*p) + (*q) |

- 3) Assumindo que o endereço de num foi atribuído a um ponteiro pnum, quais das seguintes expressões são verdadeiras?

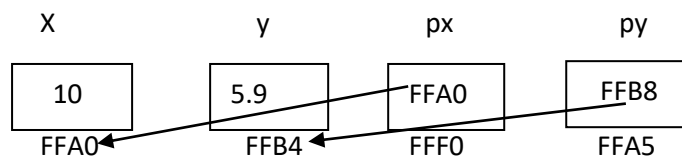
- | | |
|-----------------|-----------------|
| a) num == &pnum | b) num == *pnum |
| c) pnum == *num | d) pnum == &num |

- 4) Assumindo que o endereço da variável **x** foi atribuído a um ponteiro **px**, escreva uma expressão que não usa x e divida x por 5.

***px/5**

5) Dadas as declarações abaixo; qual é o valor dos itens:

```
int x=10, *px=&x;
float y=5.9, *py=&y;
```



- | | | |
|--------------|--------------|----------------|
| a) X = 10 | b) *py = 5.9 | c) px = FFA0 |
| d) &y = FFB4 | e) *px = 10 | f) y = 5.9 |
| g) py = FFB4 | h) &x = FFA0 | i) py++ = FFB8 |

6) Modifique o programa abaixo, corrigindo os erros existentes na passagem de parâmetros. Depois de corrigido, o que será apresentado em tela?

```
#include <stdio.h>
void função(int x, float *v1, int v2){
    *v1= (float)x/10;
    *v2 = x *10;
}

int main(){
    int num=15, mult;
    float div;
    funcao(num, mult, &div);
    printf("\n%d * 10 = %d",num, mult);
    printf("\n%d / 10 = %3.2f",num, div);
    return 0;
}

#include <stdio.h>
void funcao(int x, float *v1, int *v2){
    *v1= (float) x/10;
    *v2 = x *10;
}

int main(){
    int num=15, mult;
    float div;
    funcao(num, &div, &mult);
    printf("\n%d * 10 = %d",num, mult);
    printf("\n%d / 10 = %3.2f",num, div);
    return 0;
}
```

7) Crie uma função substring que receba duas strings e dois números inteiros (correspondentes às posições da cópia). A função deverá copiar na string2 parte da string1 de acordo com as posições.

```
char RA[9]={"AN123456"};
```

```
char curso[3];
```

Exemplo: substring(RA,curso,1,2); → a string curso receberá o conteúdo "AN"

```
char substring(char *string, char *string1, int ini, int fim)
{
    int i;
    for (i=ini;i<=fim;i++)
    {
        *string1=*(string+i-1);
        string1++;
    }
    *string1='\0';
}
```