

Exercícios sobre Ponteiros

1. O que é um ponteiro? E para que serve um ponteiro?
2. Quais das seguintes instruções são corretas para declarar um ponteiro?
  - a) `int _ptr x;`
  - b) `int *ptr;`
  - c) `*int ptr;`
  - d) `*x;`
3. Qual é a maneira correta de referenciar **ch**, assumindo que o endereço de **ch** foi atribuído ao ponteiro **indica**?
  - a) `*indica;`
  - b) `int *indica;`
  - c) `*indic;`
  - d) `ch`
  - e) `*ch;`
4. Na expressão `float *pont;` o que é do tipo `float`?
  - a) a variável `pont`.
  - b) o endereço de `pont`.
  - c) a variável apontada por `pont`.
  - d) nenhuma das anteriores.
5. 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`
6. Assumindo que queremos ler o valor de **x**, e o endereço de **x** foi atribuído a **px**, a instrução seguinte é correta? Por que?

```
scanf ( "%d", *px );
```
7. Qual é a instrução que deve ser adicionada ao programa seguinte para que ele trabalhe corretamente?

```
main ( ) {  
    int j, *pj;  
    *pj = 3;  
}
```

8. 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.

9. Qual o valor das seguintes expressões:

```
int i = 3, j = 5;
int *p = &i, *q = &j;
```

a)  $p == \&i$

b)  $*p - *q$

c)  $**\&p$

10. Qual será a saída deste programa?

```
main ( ) {
    int i = 5, *p = &i;
    printf ( "%u %d %d %d %d \n", p, *p+2, **\&p, 3**p, **\&p+4);
}
```

11. Se  $i$  e  $j$  são variáveis inteiras e  $p$  e  $q$  são ponteiros para inteiros, quais das seguintes expressões de atribuição são ilegais?

a)  $p = \&i;$

e)  $i = **\&j;$

b)  $*q = \&j;$

f)  $q = \&p;$

c)  $p = \&*i;$

g)  $i = (*p) ++ + *q;$

d)  $i = (*\&)j;$

12. Seja a seguinte seqüência de instruções em um programa 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) ao se alterar o valor de  $i$ ,  $*pti$  será modificado

e)  $pti$  é igual a 10

13. Considerando as variáveis e ponteiros definidos abaixo; quais são as atribuições permitidas?

```
int x, *ptx, **pp;
float a, *pta, **pf;
```

- |               |                |               |
|---------------|----------------|---------------|
| a) x = 100;   | e) pp = &pta;  | i) pp = &x;   |
| b) *pta = &a; | f) **pf = 7.9; | j) pf = &pta; |
| c) ptx = &a;  | g) *ptx = 20;  |               |
| d) *pf = &a;  | h) ptx = &x;   |               |

14. Considerando as variáveis e ponteiros definidos abaixo; quais são as atribuições permitidas?

```
int i, *pi, **ppi;
float f, *pf, **ppf;
```

- |                |               |                 |
|----------------|---------------|-----------------|
| a) i = f;      | e) *pf = 10;  | i) ppf = &pf;   |
| b) pf = &i;    | f) f = i;     | j) **ppi = 100; |
| c) *pf = 5.9;  | g) pi = &f;   |                 |
| d) *ppi = &pi; | h) *pi = 7.3; |                 |

15. Dadas as declarações abaixo; qual é o valor dos itens:

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

x □ FFA0	y □ FFB4	px □ FFF0	py □ FFC6	ppy □ FFA6	ppx □ FFD4
----------------	----------------	-----------------	-----------------	------------------	------------------

- |          |            |              |
|----------|------------|--------------|
| a) x =   | g) *ppx =  | m) &ppy =    |
| b) *py = | h) py =    | n) *&px =    |
| c) px =  | i) &x =    | o) **ppx++ = |
| d) &y =  | j) py++ =  | p) px++ =    |
| e) *px = | k) *px-- = | q) &ppx =    |
| f) y =   | l) **ppy = |              |