

## LISTA 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 _px;`
  - b) `int *p;`
  - c) `*int p;`
  - d) `*x;`
3. Qual 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 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);`

6. Qual é a instrução que deve ser adicionada ao programa seguinte para que ele trabalhe corretamente?  

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

7. Seja o seguinte código 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`

8. Qual será a saída deste programa supondo que **i** ocupa o endereço **4094** na memória?  

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

9. Se **i** e **j** são variáveis inteiras e **p** e **q** ponteiros para int, quais das seguintes expressões de atribuição são ilegais?

- a) `p = &i;`    b) `*q = &j;`    c) `p = &&i;`    d) `i = (*&)j;`  
 e) `i = *&j;`    f) `i = *&&j;`    g) `q = *p;`    h) `i = (*p)++ + *q`

10. O que fazem os seguintes programas?

<pre>#include &lt;conio.h&gt; #include &lt;stdio.h&gt; void main(){     int vet[] = {4,9,13};     int i;     for(i=0;i&lt;3;i++){         printf("%d ",*(vet+i));     } }</pre>	<pre>#include &lt;conio.h&gt; #include &lt;stdio.h&gt; void main(){     int vet[] = {4,9,13};     int i;     for(i=0;i&lt;3;i++){         printf("%X ",vet+i);     } }</pre>	<pre>#include &lt;conio.h&gt; #include &lt;stdio.h&gt; void main(){     int vet[] = {4,9,13};     int i;     for(i=0;i&lt;3;i++){         printf("%X ",vet+i);     } }</pre>
Programa (a)	Programa (b)	Programa (c)

9. O que faz o seguinte programa quando executado?

<pre>#include &lt;conio.h&gt; #include &lt;stdio.h&gt; void main() {     int vet[] = {4,9,12};     int i,*ptr;     ptr = vet;     for(i = 0 ; i &lt; 3 ; i++) {         printf("%d ",*ptr++);     } }</pre>	<pre>#include &lt;conio.h&gt; #include &lt;stdio.h&gt; void main(){     int vet[] = {4,9,12};     int i,*ptr;     ptr = vet;     for(i = 0 ; i &lt; 3 ; i++) {         printf("%d ",(*ptr)++);     } }</pre>
Programa (a)	Programa (b)

11. Escreva um programa que armazene os seguintes números no vetor chamado **rates**: 6.25, 6.5, 6.8, 7.2, 7.35, 7.5, 7.65, 7.8, 8.2, 8.4, 8.6, 8.8, 9.0. Indique os valores no vetor mudando o endereço em um ponteiro chamado **disp\_pt**. Use o comando **for** em seu programa.
12. Escreva um programa que armazene os seguintes números no vetor chamado **miles**: 15, 22, 16, 18, 27, 23, 20. Faça com que o programa copie os dados armazenados no **miles** para um outro vetor chamado **dist** e apresente então, os valores no vetor **dist**.
13. Assumindo que o endereço da variável **x** foi atribuído a um ponteiro **px**, escreva uma função que não use **x**, e divida **x** por 5.
14. Seja a seguinte seqüência de instruções em um programa em C:

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

Qual afirmativa é **FALSA**?

- a) **pti** armazena o endereço de **l**.
- 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.