

Lista 7**Ponteiros (Funções; Vetores e Registros)****Questão 1**

Defina (com suas palavras) o que é um ponteiro, e por que precisamos declarar o tipo de um ponteiro?

Questão 2

Se *i* é uma variável e *p* é um ponteiro que aponta para *i*, quais das seguintes expressões são apelidos (*alias*) para *i*?

- | | |
|-------------------|-------------------|
| 1. <i>*p</i> | 5. <i>*i</i> |
| 2. <i>&p</i> | 6. <i>&i</i> |
| 3. <i>*&p</i> | 7. <i>*&i</i> |
| 4. <i>&*p</i> | 8. <i>&*i</i> |

Questão 3

Qual o resultado do código abaixo?

```
1 #include <stdio.h>
2
3 int main() {
4
5     int a=1, b=10, c;
6     int *p = &a;
7     int **r = &p;
8
9     (**r)++;
10    b++;
11    c = *p + b;
12
13    printf("%d\n", c);
14
15    return 0;
16 }
```

Questão 4

Se `i` é uma variável do tipo `int` e `p` e `q` são ponteiros para `int`, quais das seguintes atribuições são corretas?

- | | |
|-----------------------------|-----------------------------|
| 1. <code>p=1;</code> | 5. <code>p = *&q</code> |
| 2. <code>*p = &i</code> | 6. <code>p = q</code> |
| 3. <code>&p = q</code> | 7. <code>*p = q</code> |
| 4. <code>p = &q</code> | 8. <code>*p = *q</code> |

Questão 5

Por que o código abaixo está errado?

```
1 void swap(int *i, int *j) {  
2     int *tmp;  
3     *tmp = *i;  
4     *i = *j;  
5     *j = *tmp;  
6 }
```

Como corrigir a função `swap`?

Questão 6

No programa abaixo, os valores de `x` e `v[0]` não deveriam ser iguais? O que acontece?

```
1 #include <stdio.h>  
2  
3 void func1 (int x) {  
4     x = 9 * x;  
5 }  
6 void func2 (int *v) {  
7     v[0] = 9 * v[0];  
8 }  
9 int main() {  
10     int x, v[2];  
11     x = 111;  
12     func1(x);  
13     printf("x: %d\n", x);  
14     v[0] = 111;  
15     func2 (v);  
16     printf("v[0]: %d\n", v[0]);  
17  
18     return 0;  
19 }
```

Questão 7

Qual é o resultado do programa abaixo?

```
1 #include <stdio.h>
2 #define N 10
3
4 int main() {
5     int i, V[N] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
6     int *p = &V[0], *q = &V[N - 1], temp;
7
8     while (p < q){
9         temp = *p;
10        *p++ = *q;
11        *q-- = temp;
12    }
13
14    p = V;
15    for(i=0; i<N; i++) printf("%d\n", *p++);
16
17    return 0;
18 }
```

Questão 8

Qual é o resultado do programa abaixo?

```
1 #include <stdio.h>
2
3 int main() {
4
5     int V[] = {5, 15, 34, 54, 14, 2, 52, 72};
6     int *p = &V[1];
7     int *q = &V[5];
8
9     printf("%d\n", *(p + 3));
10    printf("%d\n", *(q - 3));
11    printf("%d\n", q - p);
12
13    (q < p) ? printf("SIM\n"): printf("NAO\n");
14
15    if(*p < *q) printf("SIM\n");
16    else printf("NAO\n");
17
18    return 0;
19 }
```

Questão 9

Qual é o resultado do programa abaixo?

```
1 #include <stdio.h>
2
3 struct dois_valores {
4     int vi;
5     float vf;
6 };
7
8 int main() {
9     struct dois_valores reg1 = {53, 7.112}, reg2, *p = &reg1;
10    reg2.vi = (*p).vf;
11    reg2.vf = (*p).vi;
12    printf("1: %d %f\n2: %d %f\n", reg1.vi, reg1.vf, reg2.vi, reg2.vf);
13
14    return 0;
15 }
```

Questão 10

Qual é o resultado do programa abaixo?

```
1 #include <stdio.h>
2
3 struct pts_int {
4     int *pt1;
5     int *pt2;
6 };
7
8 int main(){
9
10    int i1, i2=100;
11    struct pts_int reg;
12
13    reg.pt1 = &i1;
14    reg.pt2 = &i2;
15    *reg.pt1 = -2;
16
17    printf("i1 = %d, *reg.pt1 = %d\n", i1, *reg.pt1);
18    printf("i2 = %d, *reg.pt2 = %d\n", i2, *reg.pt2);
19
20    return 0;
21 }
```