

Aula 10: Laboratório - Ponteiros (parte 2)

Prof. João Henrique Kleinschmidt

Material elaborado pelo prof. Jesús P. Mena-Chalco

3Q-2018



Prática

```
1 #include<stdio.h>
2
3 void troca(int *i, int *j) {
4     int temp;
5     temp = *i;
6     *i = *j;
7     *j = temp;
8 }
9
10 int main() {
11     int a=1;
12     int b=10;
13
14     troca(&a,&b);
15
16     printf("\ta=%d\n\tb=%d", a, b);
17 }
```

a=10

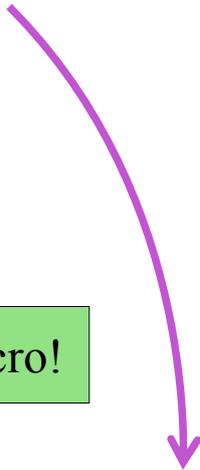
b=1

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main() {
5     int *v;
6     int n, i;
7
8     scanf( "%d", &n);
9
10    v = (int *) malloc( n*sizeof(int) );
11
12    for (i=0; i<n; i++)
13        scanf( "%d", &v[i]);
14
15    for (i=n; i>0; i--)
16        printf( "%d ", v[i-1]);
17
18    free(v);
19 }
```

Os ponteiros
facilitam a
alocação dinâmica
de memória

```
6
1
2
3
4
5
6
6 5 4 3 2 1
```

Macro!



```
1 #include <stdio.h>
2 #define max 100
3
4 void funcaoX(int M[max][max]) {
5     M[2][3] = 4;
6 }
7
8 int main() {
9     int A[max][max];
10    A[2][3] = 5;
11
12    funcaoX(A);
13
14    printf("%d", A[2][3]);
15 }
```

Escreva um programa que leia um número inteiro positivo n seguido de n números inteiros e imprima esses n números em ordem invertida.

Por exemplo, ao receber

5

22 33 44 55 66

o seu programa deve imprimir

66 55 44 33 22

- Seu programa não deve impor limitações sobre o valor de n
- Seu programa não deve usar colchetes.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main() {
5     int n, i;
6     scanf("%d", &n);
7     int *p = (int *) malloc(n*sizeof(int));
8
9     for (i=0; i<n; i++)
10         scanf("%d", p+i);
11
12     for (i=n-1; i>=0; i--)
13         printf("%d ", *(p+i));
14
15     free(p);
16 }
```

```
5
11 22 33 44 55
55 44 33 22 11
```

Os ponteiros
facilitam a
alocação dinâmica
de memória

Lista 4