

## 2 - USANDO PONTEIROS

- A linguagem C é totalmente dependente de ponteiros.
- Quando falamos de ponteiros, estamos falando em fazer manipulação lá no endereço de memória para onde o valor da variável foi alocado.
- Trabalhando com endereço de memória, estamos trabalhando a baixo nível na linguagem. Estamos fazendo acesso ao hardware lógico da memória.
- Vamos criar um programa.

```
int main(){//inicio_main
    printf("\n");

    int n;

    printf("informe um numero:\n");
    scanf("%d",&n);

    printf("O numero informado foi : %d\n",n);

    printf("Endereço de memória: %x\n",&n);|
```

```
PS C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S10(ponteiros)\2-USANDO_PONTEIROS
> cmd /c .\"p22.exe"

informe um numero:
6
O numero informado foi : 6
Endereço de memória: 61ff1c

PS C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S10(ponteiros)\2-USANDO_PONTEIROS
> cmd /c .\"p22.exe"

informe um numero:
3
O numero informado foi : 3
Endereço de memória: 61ff1c

PS C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S10(ponteiros)\2-USANDO_PONTEIROS
> |
```

- Alguns sistemas como o windows usando o mesmo espaço alocado, outros reservam um a cada execução do programa.
- inicializando um ponteiro

```
int n; //declarando_variavel - variavel que guarda seu valor na memoria

int *p; //declarando_variavel(ponteiro) - variavel que guarda um endereço de memoria.
```

- O ponteiro recebe um valor de uma variável quando passamos o endereço dessa variável para ele:

```
p = &n; //ponteiro recebe valor informado pelo usuario|
```

- O local que o SO alocar para a variável n deve ser informado para a variável ponteiro.

```
printf("Endereço do ponteiro: %p\n",p);|
```

```
informe um numero:
6
O numero informado foi : 6
Endereco de memoria: 6422296
Endereco do ponteiro: 0061FF18
```

- Primeiro valor é a forma numerica do endereço de memoria(%d)
- No segundo estamos imprimindo em hexadecimal (%p).

```
PS C:\Users\Gabri\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\300(ponteiro)\2-USANDO_PONTEIROS 20
> cmd /c .\p22.exe 21
informe um numero: 22 printf("O numero informado foi : %d\n",n);
6 23
O numero informado foi : 6 24 printf("Endereco de memoria: %d\n",&n);
Endereco de memoria: 6422296 25
Endereco do ponteiro: 6422296 26 printf("Endereco do ponteiro: %d\n",&p);
27
```

- Os dois são iguais!