



UNIVERSIDADE FEDERAL DO PIAUÍ - UFPI
CAMPUS SENADOR HELVÍDIO NUNES DE BARROS - CSHNB
CURSO DE SISTEMAS DE INFORMAÇÃO
PICOS - PI

PONTEIROS EM ESTRUTURAS DE DADOS

Prof. Ma. Luana Batista da Cruz
luana.b.cruz@nca.ufma.br

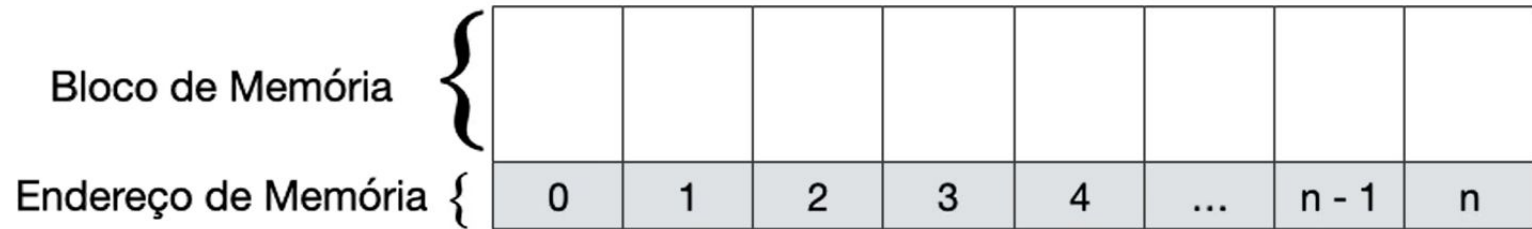
Roteiro

2

- Memória
- Conceito de ponteiros

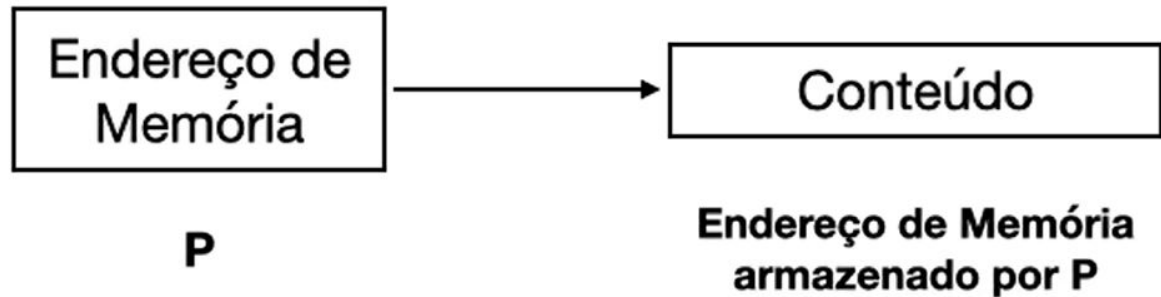
Memória

3



Conceito de ponteiros

4



Conceito de ponteiros

5

- ❑ Variável do tipo ponteiro:
 - C permite o armazenamento e a manipulação de valores de endereços de memória
 - Declaração usando tipo `*nome_do_ponteiro`
- ❑ Para cada tipo existente, há um tipo ponteiro que pode armazenar endereços de memória onde existem valores do tipo correspondente armazenados
- ❑ **Operador unário & (“endereço de”):**
 - Resulta no endereço da posição de memória reservada para a variável
- ❑ **Operador unário * (“conteúdo de”):**
 - Acessa o conteúdo do endereço de memória armazenado pela variável ponteiro

Manipulação de ponteiros em C

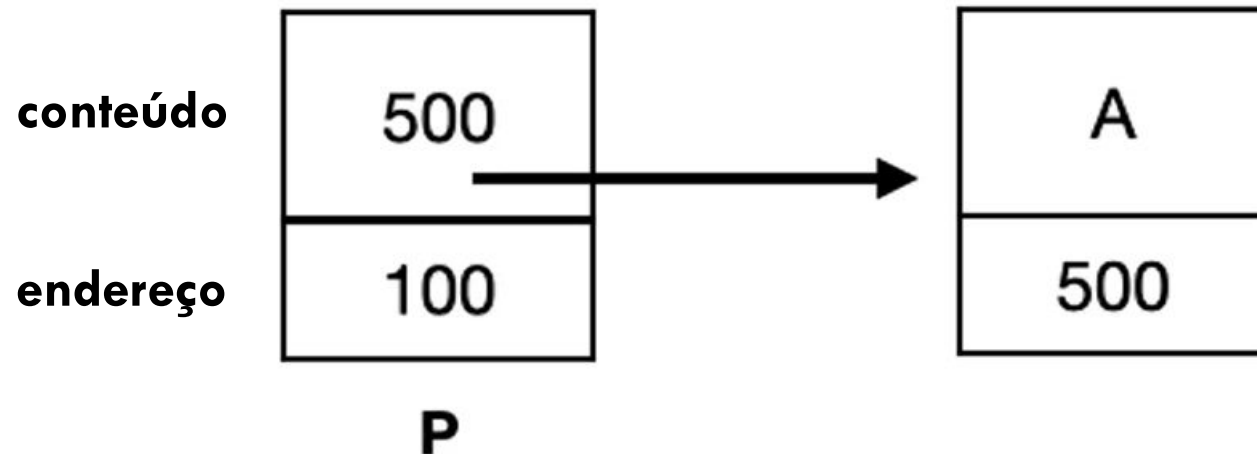
6

```
int *P, a;  
  
a = 10;  
P = NULL;  
P = (int *) malloc(sizeof(int));  
P = &a;  
  
printf("O conteúdo de P = %d", *P);
```

Manipulação de ponteiros em C

Exemplo

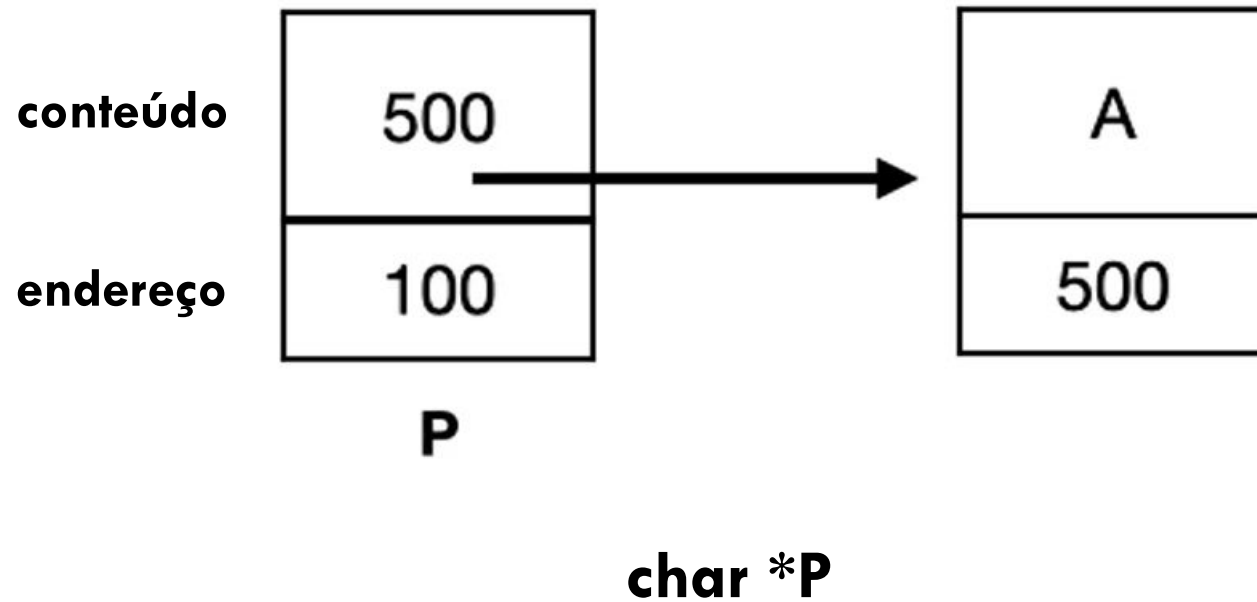
7



Manipulação de ponteiros em C

Exemplo

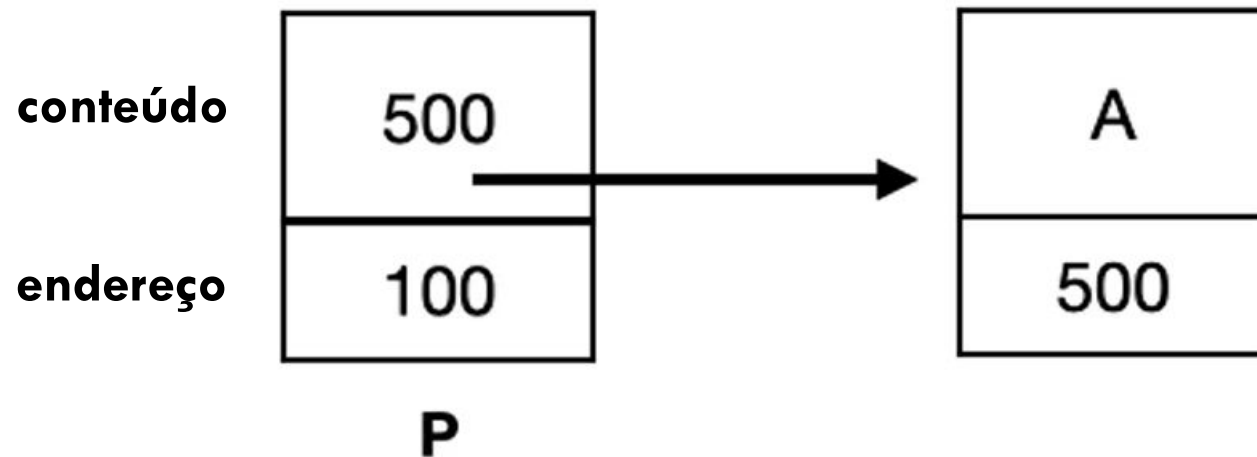
8



Manipulação de ponteiros em C

Exemplo

9

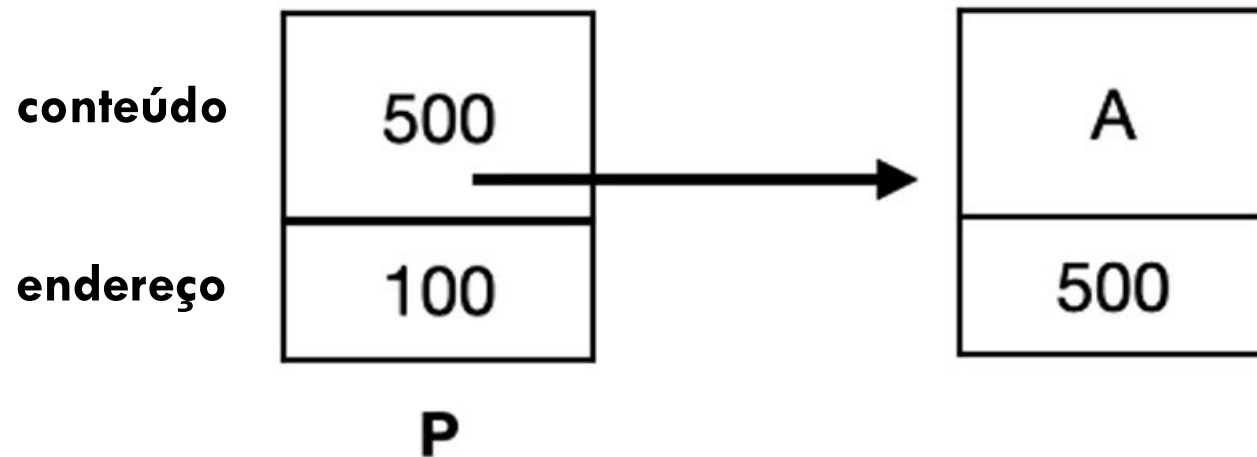


```
printf("? = %p \n",&p);  
  
printf("?? = %p \n",p);  
  
printf("??? = %c \n",*p);
```

Manipulação de ponteiros em C

Exemplo

10



```
printf("Endereço de P = %p \n",&p);
```

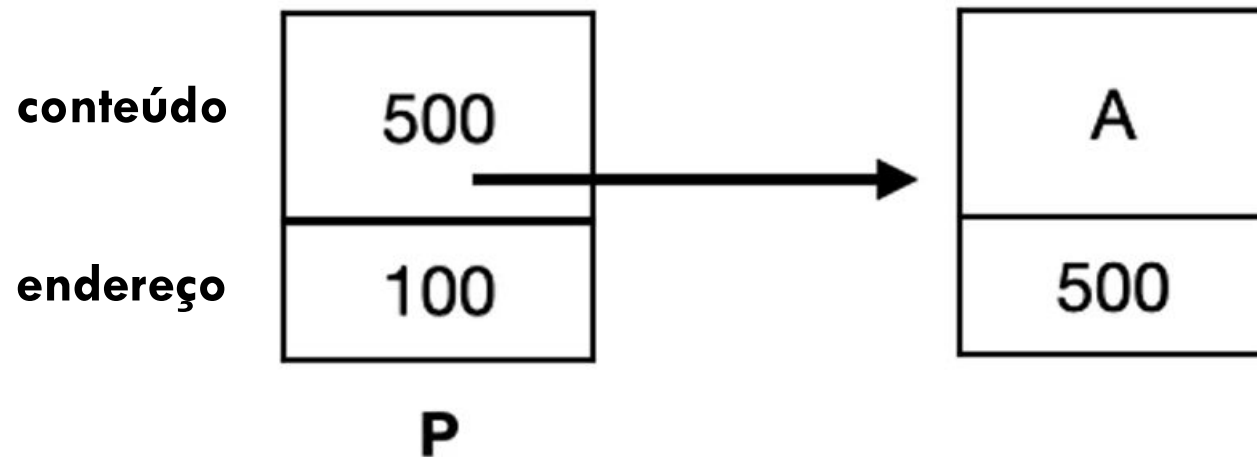
```
printf("Endereço que P aponta = %p \n",p);
```

```
printf("Conteúdo do endereço que P aponta = %c \n",*p);
```

Manipulação de ponteiros em C

Exemplo

11



Endereço de P = 100

Endereço que P aponta = 500

Conteúdo do endereço que P aponta = A

Manipulação de ponteiros em C

Exemplo

12

- ❑ `int a;`
- ❑ `int *p;`
- ❑ `int c;`

```
/* a recebe o valor 5 */  
a = 5;
```

```
/* p recebe o endereço de a  
ou seja, p aponta para a */  
p = &a;
```

```
/* posição de memória apontada por p  
recebe 6 */  
*p = 6;
```

```
/* c recebe o valor armazenado  
na posição de memória apontada por p */  
c = *p;
```

c	-	112
p	-	108
a	5	104

c	-	112
p	104	108
a	5	104

c	-	112
p	104	108
a	6	104

c	6	112
p	104	108
a	6	104

Manipulação de ponteiros em C

Exemplo

13

```
int main(){  
  
    int *P;  
    P = (int *) malloc(sizeof(int));  
    *P = 100;  
  
    novoConteudo(*P);  
    printf("Conteúdo que P aponta: %d\n", *P);  
  
    return 0;  
}
```

```
void novoConteudo(int a){  
  
    a = a + 200;  
    printf("Conteúdo de a: %d\n", a);  
  
}
```

Manipulação de ponteiros em C

Exemplo

14

```
int main(){

int *P;
P = (int *) malloc(sizeof(int));
*P = 100;

novoConteudo1(P);
printf("Conteúdo que P aponta: %d\n", *P);

return 0;
}
```

```
void novoConteudo1(int *a){

*a = *a + 200;
printf("Conteúdo de a: %d\n", *a);

}
```

Manipulação de ponteiros em C

Exemplo

15

```
int main(){

int *P;
P = (int *) malloc(sizeof(int));
*P = 100;

novoEndereco(&P);
printf("Endereço que P aponta: %p\n", P);

printf("Conteúdo que P aponta: %d\n", *P);

return 0;
}
```

```
void novoEndereco(int **a){

*a = (int *) malloc(sizeof(int));
**a = 200;

printf("Endereço que a aponta: %p\n", *a);
printf("Conteúdo do end. que a aponta:
      %d\n", **a);

}
```

Estruturas de dados

16

- ❑ Pilha
- ❑ Fila
- ❑ Listas simples
- ❑ Lista duplamente encadeada
- ❑ Listas circulares

Resumindo....

17

- Memória
- Conceitos de ponteiros
- Manipulação de ponteiros



Atividade de fixação!

18

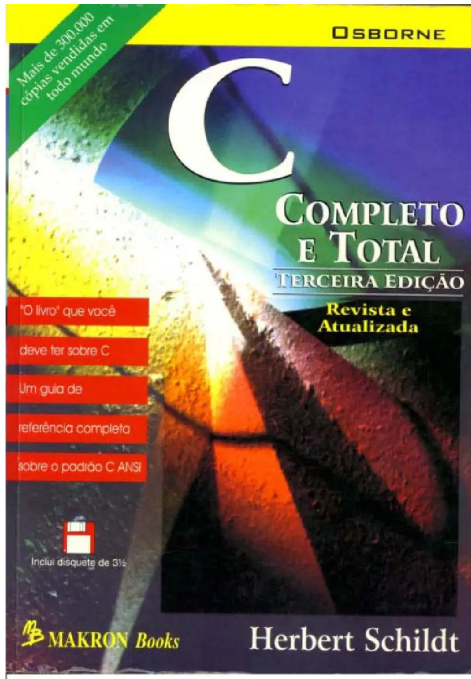
- Implementação



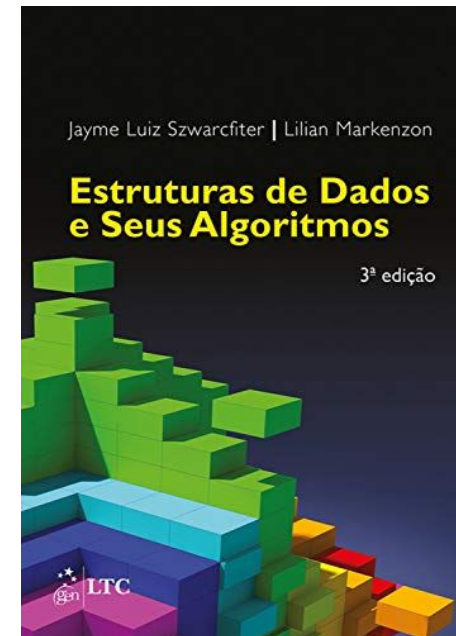
Referências



19



SCHILD, Herbert. **C completo e total**. Makron, 3ª edição revista e atualizada, 1997.



SZWARCHFITER, J. **Estruturas de Dados e seus algoritmos**. 3 ed. Rio de Janeiro: LTC, 2010.