

MC102

Algoritmos e Programação de Computadores

Aula de Laboratório 09
Instituto de Computação
Primeiro Semestre de 2012

7 de maio de 2012



Conteúdo

1 Ponteiros

2 Ordenação



Ponteiros

- São tipos especiais de dados que armazenam endereços de memória;
- **Ex.:** `int *end_a; float *end_b;`
 - `end_a` armazena endereço de memória de variáveis do tipo `int`;
 - `end_b` armazena endereço de memória de variáveis do tipo `float`;
- Existe dois operadores em ponteiros:

- 1 Operador `&` retorna o endereço de memória de uma variável;

Ex.:

```
int *end_a;  
int a = 90;  
end_a = &a;
```

- 2 Operador `*` retorna o conteúdo do endereço indicado pelo ponteiro;

Ex.:

```
printf("%d", *end_a);
```



Ponteiros

Exemplo: Operações com apontador

```
1 #include <stdio.h>
2 int main(void){
3     int a;
4     int *end_a;
5     a = 2;
6     end_a = &a; //Ponteiro aponta para o endereço de a
7     printf("\nValor de a: %d\n",a);
8     *end_a = 4; //Modifica o conteúdo do ponteiro
9     printf("\nValor de a: %d\n",a);
10    printf("\nValor de *end_a: %d\n", *end_a);
11 }
```



Ponteiros

Exemplo 2: Apontando para um vetor

```
1 #include <stdio.h>
2 int main(void){
3     int i, vetor[4] = {4,6,1,2};
4     int *end_v;
5     //Ponteiro aponta para o endereco do vetor
6     end_v = vetor;
7     for(i=0; i<4; i++)
8         printf("Valor: %d\n",end_v[i]);
9 }
```



Ponteiros

Exemplo 3: Passagem por referência

```
1 #include <stdio.h>
2
3 void somaVet(int vet[], int *s){
4     int i;
5     for(i=0; i<5; i++){
6         *s = *s + vet[i];
7     }
8 }
9
10 int main(){
11     int soma=0;
12     int vetor[5] = {6,2,3,4,7};
13     //Passa como parametro o endereco de soma
14     somaVet(vetor, &soma);
15     printf("\nValor de Soma: %d\n",soma);
16 }
```



Ponteiros - passagem por referência

Problema

Faça um programa que declare um vetor como no exemplo: `vet[10] = {7, 12, 3, 2, 4, 8, 15, 20, 30, 10}`; Utilize uma **função** para encontrar o menor valor e **passagem por referência** para a variável menor;



Ponteiros - passagem por referência

Solução

```
1 #include <stdio.h>
2 void min(int vet[], int *min){
3     int i;
4     *min = vet[0];
5     for(i=1; i<10; i++){
6         if(vet[i] < *min)
7             *min = vet[i];
8     }
9 }
10 int main(){
11     int vetor[] = {7,12,3,2,4,8,15,20,30,10};
12     int menor;
13     min(vetor, &menor);
14     printf("\nMenor: %d\n", menor);
15 }
```



Ordenação

- **Ordenação** está presente em vários problemas cotidianos;

Definição

Dado uma coleção de elementos com uma relação de ordem entre si, devemos gerar uma saída com os elementos ordenados.



Selection-Sort

- Dado um vetor, **vet**, de inteiros;
- A ideia do algoritmo é:
 - 1 Ache o menor elemento a partir da posição **0** e troque este elemento com o elemento da posição **0**;
 - 2 Ache o menor elemento a partir da posição **1** e troque este elemento com o elemento da posição **1**;
 - 3 E assim sucessivamente...



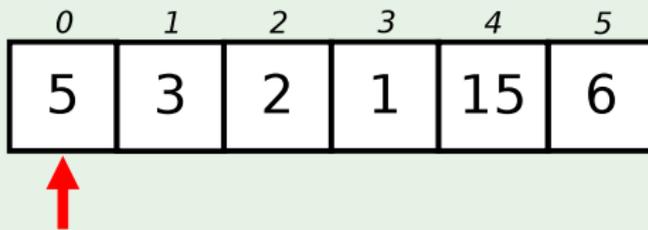
Selection-Sort

Exemplo

<i>0</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>
5	3	2	1	15	6

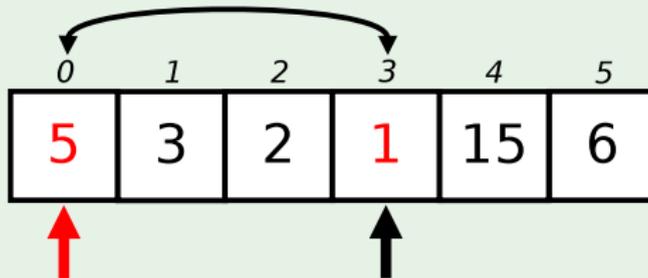
Selection-Sort

Exemplo



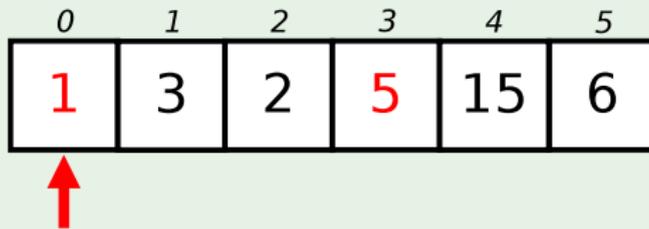
Selection-Sort

Exemplo



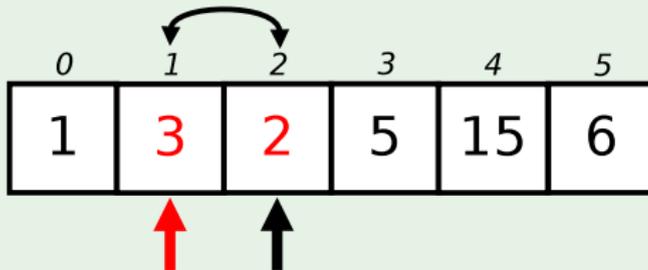
Selection-Sort

Exemplo



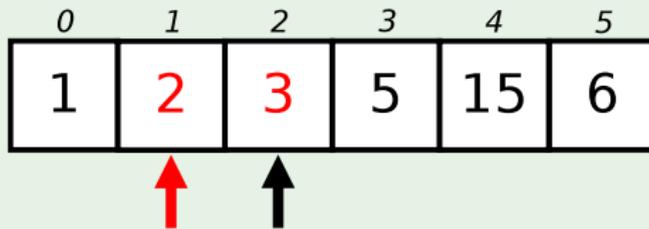
Selection-Sort

Exemplo



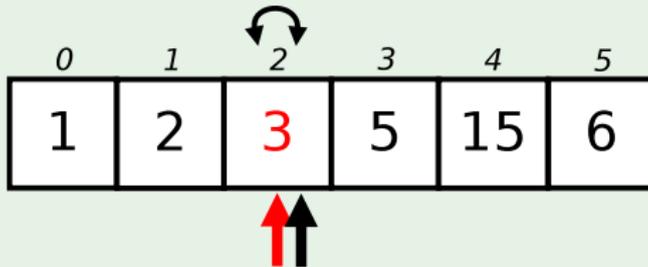
Selection-Sort

Exemplo



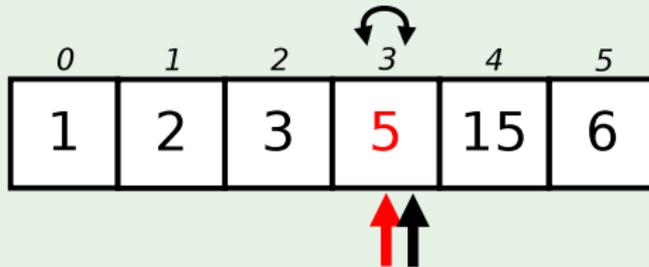
Selection-Sort

Exemplo



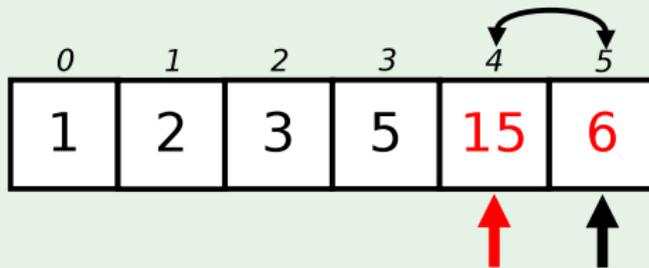
Selection-Sort

Exemplo



Selection-Sort

Exemplo



Selection-Sort

Exemplo

0	1	2	3	4	5
1	2	3	5	6	15

É necessário chegar
até a última posição?



Selection-Sort

Problema

Faça um programa que declare um vetor como no exemplo:
 $\text{vet}[10] = \{6, 12, 30, 2, 0, -5, 15, 20, 3, 10\}$; e o ordene utilizando o algoritmo **Selection-sort**.



Ordenação

Solução

```
1 #include <stdio.h>
2 void selection(int vet[], int tam){
3     int i, j, min, aux;
4     for(i = 0; i<tam-1; i++){
5         min = i;
6         for(j = i+1; j<tam; j++){
7             if(vet[min] > vet[j])
8                 min = j;
9         }
10        aux = vet[i];
11        vet[i] = vet[min];
12        vet[min] = aux;
13    }
14 }
15 int main(){
16     int i, vetor[10] = {6,12,30,2,0,-5,15,20,3,10};
17     selection(vetor, 10);
18     for(i=0; i<10; i++)
19         printf("%d ", vetor[i]);
20     printf("\n");
21 }
```



Questões?

Obrigado!

Para informação:

Página dos Laboratórios (Tarefas): <http://susy.ic.unicamp.br:9999/mc102ab>

Página do Curso: <http://www.lrc.ic.unicamp.br/~geraldoms/mc102>

E-mail:

geraldoms[at]lrc.ic.unicamp.br

brhenrique.fischer[at]gmail.com

