

Análise e Desenvolvimento de Sistemas

Manhã – ADSMA3

Estrutura de Dados

Exercício de Ordenação

Felix Petiz Bonilho RA:168048191002

Thainá Araújo Mendonça RA:1680481811044

1. Modifique o algoritmo de ordenação por contagem para que possa ordenar sequências com números repetidos.

//protótipo: void ordenaRepetidos(int origem[],int dest[],int tam)

1	3	1	1	3	2	3	1	2
---	---	---	---	---	---	---	---	---

```
void ordenaRepetidos(int origem[],int dest[],int tam) {
    int i, j, p;
    //determinar a posição de cada elemento no vetor ordenado
    for(i = 0; i < tam; i++){
        p = 0;
        for(j = 0; j < tam; j++){
            if (origem[i] > origem[j])
                p++;
            //ajusta a posição do vetor caso a variável atual já
            exista no vetor de destino
            while(dest[p] == origem[i]) p++;
            dest[p] = origem[i];
        }
    }
}
```

2. Dado o vetor

11	37	8	43	29	18	41	51	25
----	----	---	----	----	----	----	----	----

- a) Simule a execução do bubble sort até que o quinto passo (tam=5)

Resposta:

8	11	18	29	25	37	41	43	51
---	----	----	----	----	----	----	----	----

Teste de mesa:

		v[]								
tam	i	0	1	2	3	4	5	6	7	8
9	-	11	37	8	43	29	18	41	51	25
8	1	11	37	8	43	29	18	41	51	25
8	3	11	8	37	43	29	18	41	51	25
8	4	11	8	37	29	43	18	41	51	25
8	5	11	8	37	29	18	43	41	51	25
8	7	11	8	37	29	18	41	43	51	25
7	0	11	8	37	29	18	41	43	25	51
7	2	8	11	37	29	18	41	43	25	51
7	3	8	11	29	37	18	41	43	25	51

7	6	8	11	29	18	37	41	43	25	51
6	2	8	11	29	18	37	41	25	43	51
6	5	8	11	18	29	37	41	25	43	51
5	4	8	11	18	29	37	25	41	43	51
		8	11	18	29	25	37	41	43	51

b) Simule a execução do selection sort até que o quinto passo.(i=5)

Resposta:

8	11	18	25	29	37	41	51	43
---	----	----	----	----	----	----	----	----

Teste de mesa:

	num[]								
i	0	1	2	3	4	5	6	7	8
-	11	37	8	43	29	18	41	51	25
0	11	37	8	43	29	18	41	51	25
1	8	37	11	43	29	18	41	51	25
2	8	11	37	43	29	18	41	51	25
3	8	11	18	43	29	37	41	51	25
4	8	11	18	25	29	37	41	51	43
5	8	11	18	25	29	37	41	51	43
	8	11	18	25	29	37	41	51	43

c) Simule a execução do insertion sort até que o quarto passo (j=4)

Resposta:

8	11	29	37	43	18	41	51	25
---	----	----	----	----	----	----	----	----

Teste de mesa:

	v[]								
j	0	1	2	3	4	5	6	7	8
-	11	37	8	43	29	18	41	51	25
1	11	37	8	43	29	18	41	51	25

2	11	37	8	43	29	18	41	51	25
3	8	11	37	43	29	18	41	51	25
4	8	11	37	43	29	18	41	51	25
	8	11	29	37	43	18	41	51	25

d) Simule a execução do quick sort até a primeira chamada recursiva.

Resposta:

11	25	8	18	29	43	41	51	37
----	----	---	----	----	----	----	----	----

Teste de mesa:

	v[]								
i	0	1	2	3	4	5	6	7	8
-	11	37	8	43	29	18	41	51	25
0	11	37	37	43	29	18	41	51	25
1	11	25	8	43	29	18	41	51	37
2	11	25	8	43	25	18	41	51	37
3	11	25	8	18	29	43	41	51	37
4	11	25	8	18	29	43	41	51	37
	11	25	8	18	29	43	41	51	37

Na primeira chamada recursiva