

MC202 (Estruturas de Dados) - 1s2021

Tarefa 11: Mergesort e Quicksort

Prof^o Ricardo Dahab

Assistente: Elisa Dell'Arriva

Assistente: Jônatas Trabuco Belotti

Instituto de Computação - UNICAMP

Sobre a Tarefa

O objetivo desta tarefa é que o(a) aluno(a) se familiarize com algoritmos de ordenação, em especial com os de divisão e conquista. O exercício consiste em implementar três algoritmos de ordenação: mergesort, quicksort e bubblesort. A ideia é usar os algoritmos de divisão e conquista enquanto o conjunto de elementos a serem ordenados tenha tamanho no máximo um valor x . A partir de então, os números devem ser ordenados utilizando-se o algoritmo bubblesort. A cada iteração dos algoritmos de divisão e conquista deve-se imprimir o conjunto de números sendo ordenados.

Algoritmos de Divisão e Conquista

Os algoritmos de ordenação por divisão e conquista que devem ser implementados são: mergesort e quicksort. Dada uma lista de elementos L , a tarefa é ordená-la usando, primeiramente o algoritmo mergesort e, em seguida, usando o algoritmo quicksort. O pivô do algoritmo quicksort deve sempre ser o primeiro elemento da lista ou sub-lista a ser ordenada. O processo de partição (no quicksort), bem como a escolha de onde dividir a lista (no mergesort) devem ser feitos como indicado no material usado em aula.

Por serem algoritmos de divisão e conquista, a cada iteração, o tamanho das sub-listas a serem ordenadas fica menor. Quando uma sub-lista tiver tamanho menor ou igual a x , então deve-se utilizar o algoritmo bubblesort para ordenar tal sub-lista. O valor x , bem como a lista L serão dados como entrada.

Ao início de cada iteração dos algoritmos de divisão e conquista, deve ser escrita na tela a lista de valores como está no momento, ou seja, a lista antes de sofrer as eventuais trocas que tal iteração causará. Cada valor deve ser seguido de um espaço e, ao final, deve haver uma quebra de linha. Além disso, ao final, deve-se ser escrita a lista ordenada, no mesmo formato descrito acima. Antes de começar as escritas referentes ao mergesort, deve-se escrever "MERGESORT", seguido de uma quebra de linha. Analogamente, antes de começar as escritas referentes ao quicksort, deve-se escrever "QUICKSORT", seguido de uma quebra de linha. Note que não deve-se escrever a lista a cada iteração do bubblesort, pois ele não é um algoritmo de divisão e conquista.

Formato da entrada e saída

A entrada é dada por três linhas. A primeira um inteiro k , indicando o número de elementos contidos na lista que deve ser ordenada. A segunda linha consiste em k números inteiros, que são os valores a serem ordenados. Finalmente, a terceira linha contém um inteiro x , que indica o tamanho mínimo que a (sub-)lista deve ter para que ela seja ordenada utilizando os algoritmos de divisão e conquista.

Exemplo 1.

Entrada:

```
5
45 89 26 78 23
2
```

Saída:

```
MERGESORT
45 89 26 78 23
45 89 26 78 23
26 45 89 78 23
23 26 45 78 89
QUICKSORT
45 89 26 78 23
26 23 45 89 78
23 26 45 89 78
23 26 45 78 89
```

Exemplo 2.

Entrada:

```
4
4 3 2 1
2
```

Saída:

```
MERGESORT
4 3 2 1
4 3 2 1
3 4 2 1
1 2 3 4
QUICKSORT
4 3 2 1
1 3 2 4
1 2 3 4
1 2 3 4
```

Exemplo 3.

Entrada:

```
10
5 8 7 9 4 1 3 2 0 6
1
```

Saída:

MERGESORT

```
5 8 7 9 4 1 3 2 0 6
5 8 7 9 4 1 3 2 0 6
5 8 7 9 4 1 3 2 0 6
5 8 7 9 4 1 3 2 0 6
5 8 7 9 4 1 3 2 0 6
5 7 8 9 4 1 3 2 0 6
4 5 7 8 9 1 3 2 0 6
4 5 7 8 9 1 3 2 0 6
4 5 7 8 9 1 3 2 0 6
4 5 7 8 9 1 3 2 0 6
4 5 7 8 9 1 3 2 0 6
4 5 7 8 9 1 2 3 0 6
0 1 2 3 4 5 6 7 8 9
```

QUICKSORT

```
5 8 7 9 4 1 3 2 0 6
1 3 2 0 4 5 8 7 9 6
0 1 3 2 4 5 8 7 9 6
0 1 3 2 4 5 8 7 9 6
0 1 2 3 4 5 8 7 9 6
0 1 2 3 4 5 8 7 9 6
0 1 2 3 4 5 8 7 9 6
0 1 2 3 4 5 8 7 9 6
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
```

Importante

- É obrigatório implementar e utilizar os três algoritmos de ordenação mencionados neste enunciado. O não cumprimento dessa regra acarretará em nota zero nesta tarefa.
- É obrigatório liberar toda memória alocada dinamicamente. Para fazer essa verificação, recomendamos o uso da ferramenta *valgrind*. Caso haja memória não liberada, serão descontados dois pontos da nota final desta tarefa.

Alguns avisos e lembretes

- A página da disciplina no SuSy é <https://susy.ic.unicamp.br:9999/mc202abc>.
- Para submeter, utilizem somente os dígitos numéricos do RA e a senha da DAC.
- Espere, no mínimo, 60 segundos entre uma submissão e outra no SuSy, para evitar problemas na entrega.
- O número máximo de submissões é 20.
- Esta tarefa tem peso 2 (um).
- O prazo de entrega desta tarefa é 12/07/2021, às 23h59. Serão aceitas submissões atrasadas até o dia 15/07/2021 e serão aplicados os descontos de nota estabelecidos conforme o número de dias em atraso.