Lab SSC0220 – 10/10/2014

Tarefa 1

Merge Sort

Implemente uma função que receba um vetor desordenado e execute o Merge Sort sobre ele. O vetor recebido deve ser da seguinte estrutura:

```
typedef struct {
    int key;
    double value;
    int copies;
} COPIES;
```

onde `key` é uma chave de busca, `value` é um número aleatório gerado entre 0 e 1 (nesta tarefa, não será utilizado o `value`) e `copies` é um contador que indica a quantidade de vezes que o campo foi copiado para o vetor auxiliar durante o merge.

Cada posição i do vetor terá sua `key` inicialmente com valor `N – i`.

Entrada

Cada arquivo de entrada possui um ou mais casos de teste. Em cada linha há um caso de teste, composto por um inteiro N (1 <= N <= 1000000) que indica a quantidade de elementos que o vetor possui. A entrada é finalizada pelo final do arquivo (EOF).

Saída

Para cada caso de teste, imprima duas colunas. A primeira é a chave dos elementos ordenados. E a segunda é a quantidade de vezes que cada elemento mudou de lugar. Siga o formato: “%d\t%d”. Imprima uma linha em branco após o final de cada caso de teste.

Exemplo de entrada	Saída para o exemplo de entrada
10 5	1 4 2 4 3 3 4 3 5 3 6 4 7 4 8 3 9 3 10 3 1 3 2 3 3 2 4 2 5 2

Tarefa 2

Merge Sort II

Utilizando a mesma estrutura do exercício anterior, ordene o vetor em função da variável aleatória `value` que um elemento mudou de posição.

Para popular o vetor será utilizada a função rand(), gerando números no intervalo [0, 1), sendo que apenas as 6 primeiras casas após a virgula serão consideradas. De maneira a garantir que todos os vetores gerados sejam iguais, deve-se alterar a semente dos números aleatórios para o primeiro valor N que será lido pelo seu programa, utilizando a função srand().

Cada posição i do vetor terá sua `key` inicialmente com valor `N – i`.

Nesta tarefa, faça uso de uma função de comparação a ser passada como parâmetro para a função de ordenação.

Entrada

Cada arquivo de entrada possui um ou mais casos de teste. Em cada linha há um caso de teste, composto por um inteiro N (1 <= N <= 1000000) que indica a quantidade de elementos que o vetor possui. A entrada é finalizada pelo final do arquivo (EOF).

Saída

Para cada caso de teste, imprima duas colunas. A primeira é o valor dos elementos ordenados. E a segunda é a quantidade de vezes que cada elemento mudou de lugar. Siga o formato: “%lf\t%d”. Imprima uma linha em branco após o final de cada caso de teste.

Exemplo de entrada	Saída para o exemplo de entrada
10 5	0.002377 3 0.069295 3 0.128678 3 0.162307 4 0.404747 4 0.532675 3 0.573271 3 0.788725 4 0.820418 4 0.962008 3 0.364173 3 0.589623 2 0.642528 2 0.656373 2 0.662344 3