Disciplina: Estrutura de Dados II	Turma: 4EC/5CC	Data: 23/10/15	Nota:
Professor: Renato E. N. de Moraes	Semestre: 2015-2	Valor: 3,0 pt	
Aluno:	Trabalho 02 - Algoritmos de Ordenação		

Neste trabalho, faremos uma avaliação de desempenho dos algoritmos de classificação estudados em sala de aula. A primeira parte consiste em cada grupo implementar um programa em C para apresentação em laboratório da seguinte forma:

- entrada: devem ser passados como parâmetros na chamada do programa o nome de um arquivo contendo n inteiros (o arquivo deve conter um inteiro em cada linha) e um inteiro para ser usado como limite de tempo (por exemplo, se passado o valor 600, significa que qualquer algoritmo deve ser abortado quando não chegar ao final da ordenação passados 600 segundos).
- processamento: Os inteiros devem ser lidos e carregados em um vetor dinâmico. Em seguida, o vetor original deve ser classificado usando os seguintes algoritmos: InsertionSort, ShellSort, BubbleSort, SelectionSort, QuickSort, HeapSort, MergeSort e CoutingSort.
- Saída: Um arquivo texto com os inteiros ordenados. Outro arquivo texto com a quantidade de inteiros que foi ordenada, o limite de tempo usado, o tempo absoluto (em segundos) gasto por cada algoritmo na ordenação e a comparação de tempo relativa entre os algoritmos tal que o algoritmo mais rápido recebe valor $1,0$ e os demais algoritmos recebem valores relativos, por exemplo, um algoritmo que gaste o dobro de tempo quando comparado ao algoritmo mais rápido tem valor $2,0$. Exemplo de um arquivo de saída:

```
Quantidade de chaves: 30.000
Maior chave: 100.000
Limite de tempo     : 1200 segundos
Tempo absoluto dos algoritmos (ordenadamente):
CoutingSort   :  200 segundos
QuickSort     :  400 segundos
ShellSort     :  500 segundos
MergeSort     :  600 segundos
HeapSort      :  800 segundos
InsertionSort : 1000 segundos
SelectionSort : 1100 segundos
BubbleSort    :  - (abortado)
Tempo relativo entre os algoritmos (ordenadamente):
CoutingSort   :  1,0
QuickSort     :  2,0
ShellSort     :  2,5
MergeSort     :  3,0
HeapSort      :  4,0
InsertionSort :  5,0
SelectionSort :  5,5
BubbleSort    :  - (abortado)
```

O programa dessa primeira parte deve ser apresentado funcionando em uma máquina do laboratório no dia 25/11/2015 por um aluno do grupo sorteado pelo professor. Alguns arquivos de entrada serão

fornecidos pelo professor no dia da apresentação. O aluno responsável pela apresentação deverá justificar os resultados obtidos (por exemplo, quem é o mais rápido e por quê).

A segunda parte consiste em cada grupo montar um documento, impresso, contendo seis gráficos e seis tabelas (toda tabela deve correponder a um gráfico) com a descrição de como cada gráfico foi gerado, a descrição de seu conteúdo e a interpretação dos resultados (por exemplo, os resultados condizem com o esperado? Por quê?). Três gráficos devem ser plotados com os resultados absolutos de tempo (todos três gráficos desse conjunto devem ter os mesmos limites no eixo x e y). Outros três com os resultados relativos (todos três desse conjunto devem ter os mesmos limites no eixo x e y).

Para cada conjunto de três, um gráfico deve conter o pior caso, um gráfico deve conter o melhor caso e outro o caso médio. Todos os seis gráficos devem conter oito curvas (uma para cada algoritmo). Os gráficos devem ser coloridos. Os testes devem ser feitos para conjunto de chaves começando em 10.000 e crescendo em uma progressão geométrica de 4: 40.000, 160.000, 640.000...até que o algoritmo em teste gaste uma hora (3.600 segundos) para ordenar (ou estoure a memória da máquina). Nos testes, não há a necessidade de um arquivo de entrada, os números devem ser gerados aleatoriamente e salvos diretamente no vetor a ser ordenado. Todos os testes devem ser feitos em uma mesma máquina

Trabalhem em grupos de até 4 alunos. Todos os programas devem utilizar a função `clock()` do C para contagem de tempo.