



Aula Prática 07 – Ordenação: Merge e Quick Sort

- **Data de entrega: 30/05/2014 até 23:55. O que vale é o horário do Moodle, e não do seu, ou do meu, relógio!!!**

- **Decréscimo por atraso de até: 12h = 30%, 24h = 40%; 36h = 60%; 48h = 70%; Acima de 48h = 100%.**

- **Procedimento para a entrega:**

1. Crie uma pasta para a questão: **PrimeiroNome-UltimoNome-Questao** (exemplo: reinaldo-fortes-1).
 2. Crie os arquivos de código fonte necessários para a solução da questão (arquivos **.c** e **.h**)
 3. Compile na linha de comando usando **gcc *.c -o prog.exe -lm**.
 4. Execute usando redirecionamento de entrada: **./prog.exe < entrada.txt**.
 5. Apague os arquivos gerados na compilação e o arquivo de entrada (**mantenha apenas os arquivos .c e .h**).
 6. Compacte a pasta criada no item 1.
 7. Faça a entrega do arquivo compactado no moodle, na tarefa destinada à prática e questão correspondentes.
- A cada etapa, verifique se o resultado está conforme o esperado.
 - **Não utilize caracteres acentuados ou especiais para nomes de pastas e arquivos.**

- **Bom trabalho!**

Questão 01

Implemente os algoritmos de ordenação: MergeSort e QuickSort. Cada algoritmo receberá apenas um vetor de inteiros e o tamanho deste vetor e retornará o número de comparações de chaves e movimentos de registros realizados durante sua execução.

Exemplo de entrada e Saída

A entrada inicia com o número de vetores a serem ordenados. Cada vetor é iniciado pelo número de elementos contidos no vetor e é seguido dos elementos a serem ordenados.

A saída apresenta os registros ordenados e a quantidade de comparações e movimentos realizados na execução de cada algoritmo, seguindo o padrão definido no exemplo a seguir.

A saída deve incluir os resultados dos algoritmos implementados na *Prática 06*.

Entrada	Saida
2 10 1 2 3 4 5 6 7 8 9 10 10 5 4 3 2 1 6 7 8 9 10	Vetor 1: - Ordenacao: - - Bubble.....: 1 2 3 4 5 6 7 8 9 10 - - Selection.: 1 2 3 4 5 6 7 8 9 10 - - Insertion.: 1 2 3 4 5 6 7 8 9 10 - - Merge.....: 1 2 3 4 5 6 7 8 9 10 - - Quick.....: 1 2 3 4 5 6 7 8 9 10 - Comparacoes Movimentacoes: - - Bubble.....: C11 M11 - - Selection.: C12 M12 - - Insertion.: C13 M13 - - Merge.....: C14 M14 - - Quick.....: C15 M15 Vetor 2: - Ordenacao: - - Bubble.....: 1 2 3 4 5 6 7 8 9 10 - - Selection.: 1 2 3 4 5 6 7 8 9 10 - - Insertion.: 1 2 3 4 5 6 7 8 9 10 - - Merge.....: 1 2 3 4 5 6 7 8 9 10 - - Quick.....: 1 2 3 4 5 6 7 8 9 10 - Comparacoes Movimentacoes: - - Bubble.....: C21 M21 - - Selection.: C22 M22 - - Insertion.: C23 M23 - - Merge.....: C24 M24 - - Quick.....: C25 M25

Os valores Cxy e Mxy correspondem aos valores retornados pelas funções de ordenação para o número de comparações de chaves e o número de movimentações de registros, respectivamente. Tais valores não serão apresentados aqui, pois podem variar de uma implementação para outra.