

Relatório Projeto 4.2 AED 2020/2021

Nome: Pedro Afonso Ferreira Lopes Martins

Nº Estudante: 2019216826

TP (inscrição): PL8

Login no Mooshak: 2019216826

Nº de horas de trabalho: 08H

Aulas Práticas de Laboratório: 02H

Fora de Sala de Aula: 06H

(A Preencher pelo Docente) CLASSIFICAÇÃO:

Comentários:

Registrar os tempos computacionais do QS e das 4 variantes selecionadas do QS+IS para os diferentes tipos de sequências. O tamanho das sequências (N) deve ser crescente e terminar em 10,000,000. Só deve ser contabilizado o tempo de ordenamento. Exclui-se o tempo de leitura do input e de impressão dos resultados. Devem apresentar e discutir as regressões para a melhor variante em cada tipo de sequência.

Gráfico para ALEATORIA

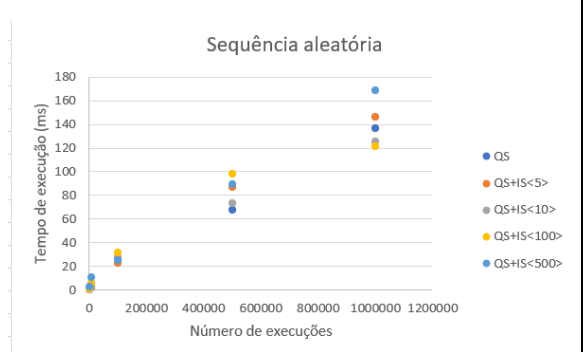


Gráfico para ORDENADA_DECRESCENTE

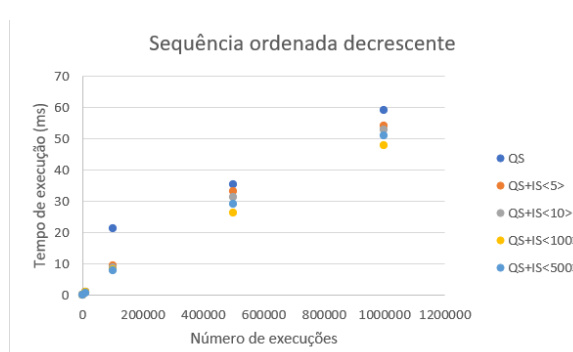


Gráfico para QUASE_ORDENADA_1%

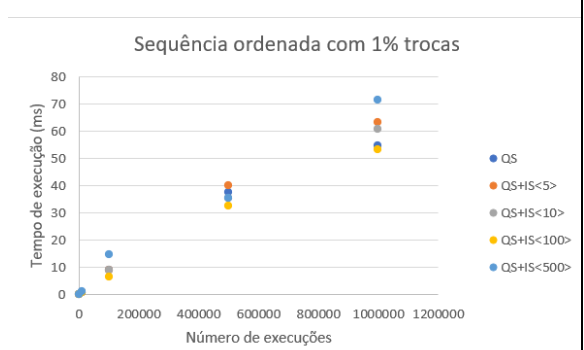
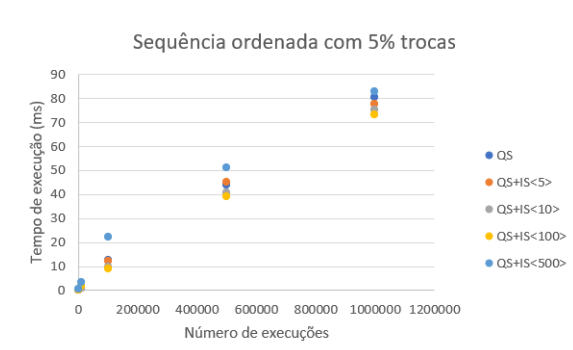


Gráfico para QUASE_ORDENADA_5%



Análise dos resultados:

Ao analisar os 4 gráficos constata-se que todos se encontram relativamente próximos e com o pior caso abaixo de 100ms, com a exceção da sequência aleatória. A ordenada decrescente e o ordenado com 1% trocas são os melhores gráficos, com valor para 1.000.000 perto de 50-60ms, ganhos enormes comparativamente ao Shell Sort. A ordenada com 5% trocas mantém-se pior que a 1% de trocas, visto que irá realizar um número superior de mudanças de números, reduzindo a ordem da sequência e necessitando assim de mais iterações e chamadas

recursivas para atingir a ordenação. Relativamente aos algoritmos, verificou-se que o QS+IS<100> é a melhor variante, atingindo tempos superiores em ~80% dos testes efetuados. Também é possível constatar que quanto maior é o k (k relativo a QS+IS< k >), e quanto menor é o número de elementos no array, pior serão os tempos, uma vez que irão ser realizados muitos Insertion Sorts para um número reduzido de array, o que leva a uma quebra de performance pela simples ineficácia deste algoritmo comparativamente ao quick sort, não justificando a sua utilização. Regra geral, o QS+IS<500> é o pior algoritmo na maioria dos casos, uma vez que são realizados demasiados Insertion Sorts nos subarrays, o que leva a tempos de execução acima das outras variantes e Quick Sort sem Insertion Sort.