

Trabalho 1º Bimestre: Métodos de Ordenação

Projeto e Análise de Algoritmos - PAA

2023

André Luiz Brun¹

¹Colegiado de Ciência da Computação
Campus de Cascavel - UNIOESTE

***Resumo.** Este documento consiste na especificação formal do primeiro trabalho da disciplina de Projeto e Análise de Algoritmos (Csc2152) para o ano letivo de 2023. Aqui são apresentadas as atividades a serem desenvolvidas e como cada processo deverá ser realizado. Além disso, o documento contém as informações sobre a formação das equipes, o objeto de trabalho de cada uma e as datas de entrega e apresentação dos relatórios.*

1. Introdução

O objetivo do primeiro trabalho da disciplina consiste em comparar o comportamento, em termos de acurácia, de classificadores baseados em diferentes conceitos sobre uma mesma base de dados. Além disso, pretende-se comparar algumas estratégias de combinação desses classificadores e analisar se a adoção da estratégia de múltiplos classificadores leva a melhores taxas de acerto.

2. Vinicius Drage, Gustavo Martini e Gustavo Macedo

2.1. Objetivo

Aplicar e corroborar conceitos adquiridos com relação ao método quadrático BubbleSort e analisar a possibilidade de otimização do funcionamento básico do algoritmo.

2.2. Tarefa

Implementar as seguintes estratégias de ordenação do BubbleSort:

1. Abordagem clássica em que o método tem todas as iterações executadas normalmente, sem nenhum tipo de otimização;
2. Abordagem com uma flag que identifica se foi realizada alguma operação de swap e, em caso negativo, termina a execução do método sem a necessidade de executar as iterações restantes.

Deve-se avaliar o comportamento dos métodos perante conjuntos de testes com diferentes características. Durante a execução dos testes o critério de análise será o **tempo cronológico** gasto para a execução dos métodos.

2.3. Como

- A linguagem utilizada no desenvolvimento é de vossa escolha. Entretanto, a mesma linguagem deve ser adotada para todos os métodos. Além disso, deve-se empregar as mesmas estratégias para a realização do trabalho. Por exemplo, empregar soluções recursivas para todos ou nenhum deles;
- A forma com que os métodos serão implementados é determinada pelo grupo;
- A entrada dos dados deve ser feita com base nos arquivos texto disponíveis nos links a seguir:
Aleatórios
Decrescentes
Crescentes
Parcialmente ordenados

Devem ser construídos quatro conjuntos de testes, conforme os arquivos disponíveis nos links apresentados acima:

1. Os dois métodos de ordenação trabalhando sobre conjunto aleatórios;
2. Os dois métodos de ordenação trabalhando sobre conjuntos ordenados de forma decrescente;
3. Todos os métodos sobre conjuntos ordenados de forma crescente;
4. Todos os métodos sobre conjuntos parcialmente ordenados de forma crescente.

Além disso, deve ser analisado o comportamento de cada técnica sobre as diferentes entradas conforme descrito a seguir:

1. O comportamento do BubbleSort clássico sobre os quatro cenários de entrada;
2. O comportamento do BubbleSort otimizado sobre os quatro cenários de entrada;

3. Gabriel Norato Claro, Gabriel Mazzuco e Rodrigo Rocha

3.1. Objetivo

Aplicar e corroborar conceitos adquiridos com relação aos métodos de ordenação quadráticos, bem como ilustrar os comportamentos apresentados pelas estratégias em diferentes cenários.

3.2. Tarefa

Implementar as seguintes estratégias de ordenação:

1. BubbleSort;
2. InsertionSort;
3. SelectionSort.

Deve-se avaliar o comportamento dos métodos perante conjuntos de testes com diferentes características. Durante a execução dos testes o critério de análise será o **tempo cronológico** gasto para a execução dos métodos.

3.3. Como

- A linguagem utilizada no desenvolvimento é de vossa escolha. Entretanto, a mesma linguagem deve ser adotada para todos os métodos. Além disso, deve-se empregar as mesmas estratégias para a realização do trabalho. Por exemplo, empregar soluções recursivas para todos ou nenhum deles;
- A forma com que os métodos serão implementados é determinada pelo grupo;
- A entrada dos dados deve ser feita com base nos arquivos texto disponíveis nos links a seguir:

Aleatórios

Decrescentes

Crescentes

Parcialmente ordenados

Devem ser construídos quatro conjuntos de testes, conforme os arquivos disponíveis nos links apresentados acima:

1. Os três métodos de ordenação trabalhando sobre conjunto aleatórios;
2. Os três métodos de ordenação trabalhando sobre conjuntos ordenados de forma decrescente;
3. Todos os métodos sobre conjuntos ordenados de forma crescente;
4. Todos os métodos sobre conjuntos parcialmente ordenados de forma crescente.

Além disso, deve ser analisado o comportamento de cada técnica sobre as diferentes entradas conforme descrito a seguir:

1. BubbleSort sobre os quatro tipos de entrada;
2. SelectionSort sobre os quatro tipos de entrada;
3. InsertionSort sobre os quatro tipos de entrada.

4. Gabriel Andrade de Araujo, Guilherme Marchiotti Ferraz e Roberval Requião Junior

4.1. Objetivo

Aplicar e corroborar conceitos adquiridos com relação aos métodos de ordenação de ordem $O(n \log_2 n)$, bem como ilustrar os comportamentos apresentados pelas estratégias em diferentes cenários.

4.2. Tarefa

Implementar as seguintes estratégias de ordenação:

1. MergeSort;
2. QuickSort;
3. HeapSort.

Deve-se avaliar o comportamento dos métodos perante conjuntos de testes com diferentes características. Durante a execução dos testes o critério de análise será o **tempo cronológico** gasto para a execução dos métodos.

4.3. Como

- A linguagem utilizada no desenvolvimento é de vossa escolha. Entretanto, a mesma linguagem deve ser adotada para todos os métodos. Além disso, deve-se empregar as mesmas estratégias para a realização do trabalho. Por exemplo, empregar soluções recursivas para todos ou nenhum deles;
- A forma com que os métodos serão implementados é determinada pelo grupo;
- A entrada dos dados deve ser feita com base nos arquivos texto disponíveis nos links a seguir:

Aleatórios

Decrescentes

Crescentes

Parcialmente ordenados

Devem ser construídos quatro conjuntos de testes, conforme os arquivos disponíveis nos links apresentados acima:

1. Os três métodos de ordenação trabalhando sobre conjunto aleatórios;
2. Os três métodos de ordenação trabalhando sobre conjuntos ordenados de forma decrescente;
3. Todos os métodos sobre conjuntos ordenados de forma crescente;
4. Todos os métodos sobre conjuntos parcialmente ordenados de forma crescente.

Além disso, deve ser analisado o comportamento de cada técnica sobre as diferentes entradas conforme descrito a seguir:

1. QuickSort sobre os quatro tipos de entrada;
2. MergeSort sobre os quatro tipos de entrada;
3. HeapSort sobre os quatro tipos de entrada.

5. Heloisa Aparecida Alves, Eduarda Elger e Ellen Carine Bonafin Marques

5.1. Objetivo

Aplicar e corroborar conceitos adquiridos com relação ao método de ordenação MergeSort, bem como ilustrar o comportamento apresentado pela estratégia em diferentes cenários.

5.2. Tarefa

Implementar as seguintes estratégias de ordenação:

1. MergeSort clássico em que a cada iteração o problema é dividido ao meio e duas novas instâncias do problema são chamadas;
2. Um MergeSort diferenciado, chamado SuperMerge. Nesta abordagem o problema será dividido em quatro subpartes e para cada uma é então executada uma nova chamada recursiva.

Deve-se avaliar o comportamento dos métodos perante conjuntos de testes com diferentes características. Durante a execução dos testes o critério de análise será o **tempo cronológico** gasto para a execução dos métodos.

5.3. Como

- A linguagem utilizada no desenvolvimento é de vossa escolha. Entretanto, a mesma linguagem deve ser adotada para todos os métodos. Além disso, deve-se empregar as mesmas estratégias para a realização do trabalho.
- A forma com que os métodos serão implementados é determinada pelo grupo;
- A entrada dos dados deve ser feita com base nos arquivos texto disponíveis nos links a seguir:
Aleatórios
Decrescentes
Crescentes
Parcialmente ordenados

Devem ser construídos quatro conjuntos de testes, conforme os arquivos disponíveis nos links apresentados acima:

1. Os dois métodos de ordenação trabalhando sobre conjunto aleatórios;
2. As duas estratégias de ordenação trabalhando sobre conjuntos ordenados de forma decrescente;
3. Todos os métodos sobre conjuntos ordenados de forma crescente;
4. Todos os métodos sobre conjuntos parcialmente ordenados de forma crescente.

Além disso, deve ser analisado o comportamento de cada técnica sobre as diferentes entradas conforme descrito a seguir:

1. MergeSort clássico sobre os quatro tipos de entrada;
2. SuperMerge sobre os quatro tipos de entrada;

6. Jaqueline Cavaller Faino, Davi Marchetti Giacomel e Maria Eduarda Crema Carlos

6.1. Objetivo

Aplicar e corroborar conceitos adquiridos com relação ao método de ordenação BucketSort combinado com o algoritmo InsertionSort, bem como ilustrar o comportamento apresentado pela estratégia em diferentes cenários.

6.2. Tarefa

Implementar as seguintes estratégias de ordenação:

1. InsertionSort tradicional;
2. Abordagem alternativa do InsertionSort usando a estratégia de BucketSort, ou seja, o conjunto de dados dos vetores de entrada devem ser particionados em baldes e então ordenados pelo método de InsertionSort. 1. Para tanto, deverão ser testadas quatro abordagens distintas da técnica implementada, cada uma com uma quantidade diferente de baldes, a saber:

- (a) 10 baldes
- (b) 100 baldes
- (c) 1000 baldes.

Deve-se avaliar o comportamento dos métodos perante conjuntos de testes com diferentes características. Durante a execução dos testes o critério de análise será o **tempo cronológico** gasto para a execução dos métodos.

6.3. Como

- A linguagem utilizada no desenvolvimento é de vossa escolha. Entretanto, a mesma linguagem deve ser adotada para todos os métodos. Além disso, deve-se empregar as mesmas estratégias para a realização do trabalho.
- Os pontos de segmentação entre os baldes é de escolha da equipe;
- A forma com que os métodos serão implementados é determinada pelo grupo;
- A entrada dos dados deve ser feita com base nos arquivos texto disponíveis nos links a seguir:
Aleatórios
Decrescentes
Crescentes
Parcialmente ordenados

Devem ser construídos quatro conjuntos de testes, conforme os arquivos disponíveis nos links apresentados acima:

1. Os quatro métodos de ordenação trabalhando sobre conjunto aleatórios;
2. As quatro estratégias de ordenação trabalhando sobre conjuntos ordenados de forma decrescente;
3. Todos os métodos sobre conjuntos ordenados de forma crescente;
4. Todos os métodos sobre conjuntos parcialmente ordenados de forma crescente.

Além disso, deve ser analisado o comportamento de cada técnica sobre as diferentes entradas conforme descrito a seguir:

1. InsertionSort clássico sobre os quatro tipos de entrada;
2. InsertionSort combinando com 10 baldes sobre os quatro tipos de entrada;
3. InsertionSort combinando com 100 baldes sobre os quatro tipos de entrada;
4. InsertionSort combinando com 1000 baldes sobre os quatro tipos de entrada;

7. Gabriel Santos Ramos, Lucas Becker da Conceição e Rafael Gotz

7.1. Objetivo

Aplicar e corroborar conceitos adquiridos com relação ao método de ordenação Quicksort $O(n \log_2 n)$ em comparação a um método de ordenação de ordem linear $O(n + k)$.

7.2. Tarefa

Implementar as seguintes estratégias de ordenação:

1. QuickSort tradicional;
2. CountingSort;

Deve-se avaliar o comportamento dos métodos perante conjuntos de testes com diferentes características. Durante a execução dos testes o critério de análise será o **tempo cronológico** gasto para a execução dos métodos.

7.3. Como

- A linguagem utilizada no desenvolvimento é de vossa escolha. Entretanto, a mesma linguagem deve ser adotada para todos os métodos. Além disso, deve-se empregar as mesmas estratégias para a realização do trabalho;
- Para a escolha do pivô do QuickSort devem ser sorteados três elementos aleatórios do conjunto e escolhida a mediana deste conjunto como pivô para aquela etapa do método. Se o vetor tiver menos que 3 elementos, usar um elemento escolhido aleatoriamente;
- **Durante a execução do CountingSort para se descobrir qual o maior valor presente no conjunto de entrada deve-se fazer uma varredura no conjunto. Essa varredura deve entrar na contagem de tempo do método;**
- A forma com que os métodos serão implementados é determinada pelo grupo;
- A entrada dos dados deve ser feita com base nos arquivos texto disponíveis nos links a seguir:
 - Aleatórios
 - Decrescentes
 - Crescentes
 - Parcialmente ordenados

Devem ser construídos quatro conjuntos de testes, conforme os arquivos disponíveis nos links apresentados acima:

1. Os dois métodos de ordenação trabalhando sobre conjunto aleatórios;
2. As duas estratégias de ordenação trabalhando sobre conjuntos ordenados de forma decrescente;
3. Todos os métodos sobre conjuntos ordenados de forma crescente;
4. Todos os métodos sobre conjuntos parcialmente ordenados de forma crescente.

Além disso, deve ser analisado o comportamento de cada técnica sobre as diferentes entradas conforme descrito a seguir:

1. QuickSort clássico sobre os quatro tipos de entrada;
2. CountingSort sobre os quatro tipos de entrada;

8. Alonso Lucca Fritz e Lucca Abbado Neres

8.1. Objetivo

Aplicar e corroborar conceitos adquiridos com relação ao método de ordenação QuickSort, bem como ilustrar o comportamento apresentado pela estratégia em diferentes cenários.

8.2. Tarefa

Implementar as seguintes estratégias de ordenação:

1. QuickSort (Randômico) em que a escolha do pivô é feita de forma totalmente aleatória;
2. QuickSort (Central) em que a escolha do pivô é feita selecionando-se o elemento na posição central do vetor;
3. QuickSort (Mediana) em que para se escolher o pivô são sorteados três elementos do vetor e o pivô será o elemento central dos três valores sorteados.

Deve-se avaliar o comportamento dos métodos perante conjuntos de testes com diferentes características. Durante a execução dos testes o critério de análise será o **tempo cronológico** gasto para a execução dos métodos.

8.3. Como

- A linguagem utilizada no desenvolvimento é de vossa escolha. Entretanto, a mesma linguagem deve ser adotada para todos os métodos. Além disso, deve-se empregar as mesmas estratégias para a realização do trabalho.
- A forma com que os métodos serão implementados é determinada pelo grupo;
- A entrada dos dados deve ser feita com base nos arquivos texto disponíveis nos links a seguir:
Aleatórios
Decrescentes
Crescentes
Parcialmente ordenados

Devem ser construídos quatro conjuntos de testes, conforme os arquivos disponíveis nos links apresentados acima:

1. As três estratégias de ordenação trabalhando sobre conjuntos ordenados de forma decrescente;
2. Os três métodos de ordenação trabalhando sobre conjunto aleatórios;
3. Todos os métodos sobre conjuntos ordenados de forma crescente;
4. Todos os métodos sobre conjuntos parcialmente ordenados de forma crescente.

Além disso, deve ser analisado o comportamento de cada técnica sobre as diferentes entradas conforme descrito a seguir:

1. QuickSort Randômico sobre os quatro tipos de entrada;
2. QuickSort Central sobre os quatro tipos de entrada;
3. QuickSort Mediana sobre os quatro tipos de entrada;

9. Gustavo Henrique Marchezoni Orlandini, Lucas David Tomalack de Souza e Rafael Roberto Hoffmann

9.1. Objetivo

Aplicar e corroborar conceitos adquiridos com relação ao método de ordenação MergeSort, bem como elucidar se a passagem por referência, trabalhando com um único vetor em memória é mais vantajosa que criar uma cópia do vetor a cada chamada recursiva do algoritmo.

9.2. Tarefa

Implementar as seguintes estratégias de ordenação:

1. Abordagem clássica do MergeSort que aloca um vetor temporário a cada chamada recursiva (solução vista em aula);
2. Abordagem alternativa do MergeSort que, ao invés de instanciar um novo vetor a cada chamada, simplesmente sobrepõe o vetor de entrada (que foi passado por referência).

Deve-se avaliar o comportamento dos métodos perante conjuntos de testes com diferentes características. Durante a execução dos testes o critério de análise será o **tempo cronológico** gasto para a execução dos métodos.

9.3. Como

- A linguagem utilizada no desenvolvimento é de vossa escolha. Entretanto, a mesma linguagem deve ser adotada para todos os métodos. Além disso, deve-se empregar as mesmas estratégias para a realização do trabalho.
- A forma com que os métodos serão implementados é determinada pelo grupo;
- A entrada dos dados deve ser feita com base nos arquivos texto disponíveis nos links a seguir:
Aleatórios
Decrescentes
Crescentes
Parcialmente ordenados

Devem ser construídos quatro conjuntos de testes, conforme os arquivos disponíveis nos links apresentados acima:

1. As duas estratégias de ordenação trabalhando sobre conjuntos ordenados de forma decrescente;
2. Os dois métodos de ordenação trabalhando sobre conjunto aleatórios;
3. Todos os métodos sobre conjuntos ordenados de forma crescente;
4. Todos os métodos sobre conjuntos parcialmente ordenados de forma crescente.

Além disso, deve ser analisado o comportamento de cada técnica sobre as diferentes entradas conforme descrito a seguir:

1. MergeSort com alocação de novo vetor sobre os quatro tipos de entrada;

2. MergeSort utilizando apenas passagem por referência Central sobre os quatro tipos de entrada.

10. Arthur Augusto Eich e Gustavo Pauli da Luz

10.1. Objetivo

Aplicar e corroborar conceitos adquiridos com relação ao método de ordenação QuickSort analisando-se seu desempenho quando implementado de forma recursiva frente à uma implementação iterativa.

10.2. Tarefa

Implementar as seguintes estratégias de ordenação:

1. QuickSort recursivo;
2. QuickSort iterativo.

Deve-se avaliar o comportamento dos métodos perante conjuntos de testes com diferentes características. Durante a execução dos testes o critério de análise será o **tempo cronológico** gasto para a execução dos métodos.

10.3. Como

- A linguagem utilizada no desenvolvimento é de vossa escolha. Entretanto, a mesma linguagem deve ser adotada para todos os métodos. Além disso, deve-se empregar as mesmas estratégias para a realização do trabalho.
- A forma com que os métodos serão implementados é determinada pelo grupo;
- Para a determinação do pivô deve-se escolher o elemento central do vetor;
- A entrada dos dados deve ser feita com base nos arquivos texto disponíveis nos links a seguir:
 - Aleatórios
 - Decrescentes
 - Crescentes
 - Parcialmente ordenados

Devem ser construídos quatro conjuntos de testes, conforme os arquivos disponíveis nos links apresentados acima:

1. As duas estratégias de ordenação trabalhando sobre conjuntos ordenados de forma decrescente;
2. Os dois métodos de ordenação trabalhando sobre conjunto aleatórios;
3. Todos os métodos sobre conjuntos ordenados de forma crescente;
4. Todos os métodos sobre conjuntos parcialmente ordenados de forma crescente.

Além disso, deve ser analisado o comportamento de cada técnica sobre as diferentes entradas conforme descrito a seguir:

1. QuickSort iterativo sobre os quatro tipos de entrada;
2. QuickSort recursivo sobre os quatro tipos de entrada.

11. Relatório

Deve ser elaborado um relatório técnico em formato pdf contendo:

- o Descrição de como foi realizado o processo empírico de determinação dos custos: cenário de realização dos experimentos e como foram tomadas as métricas exigidas.
 - Detalhar a configuração usada nos testes (Processador, SO, IDE, etc..).
- Gráficos evidenciando o comportamento dos métodos perante todos os cenários considerando o tamanho dos conjuntos de entrada.
- o Análise do comportamento dos métodos durante a execução dos testes.
 - Esta análise deve ser feita com bastante critério e ser esclarecedora, apontando razões para os comportamentos observados.
 - **AS figuras e gráficos devem ser invocadas no texto e explicadas.**

O formato do relatório deve ser a formatação presente neste texto. As regras para tal podem ser obtidas no link download. No arquivo disponível pode-se utilizar a formatação em arquivo .doc ou em latex.

12. Código-fonte

Além do relatório citado, cada equipe deverá enviar os códigos fonte construídos para a execução dos experimentos. Ambos arquivos podem ser compactados e enviados como arquivo único.

Não é necessário o envio do projeto compilado, apenas do código fonte base.

Neste código devem constar os comandos utilizados para tomadas de tempo de execução e contagem das variáveis de interesse.

13. Para quando?

O trabalho deverá ser submetido no link disponibilizado na turma de disciplina dentro do ambiente Microsoft Teams até as **23:59 do dia xx/10/2023**.

As apresentações serão realizadas na aula do dia **xx/10/2023**.

Cada grupo terá 15 minutos para apresentar o trabalho realizado, focando na descrição do problema, nos desempenhos obtidos e no resultado da análise estatística. **A qualidade da apresentação terá influência na nota do trabalho.**