



UNIVERSIDADE DE BRASÍLIA

BRASÍLIA

GENILSON SILVA DE ARAÚJO JUNIOR

JOÃO LUCAS MIRANDA DE SOUSA

DANIELA PINHEIRO BARBOSA

RENANN DE OLIVEIRA GOMES

HENRIQUE TORRES LANDIN

RELATÓRIO DO TRABALHO PRÁTICO DE ESTRUTURA DE DADOS I:

BUSCA E ORDENAÇÃO

BRASÍLIA-DF

JANEIRO/2023

GENILSON SILVA DE ARAÚJO JUNIOR

JOÃO LUCAS MIRANDA DE SOUSA

DANIELA PINHEIRO BARBOSA

RENANN DE OLIVEIRA GOMES

HENRIQUE TORRES LANDIN

RELATÓRIO DO TRABALHO PRÁTICO DE ESTRUTURA DE DADOS I:

BUSCA E ORDENAÇÃO

Trabalho prático 1, traçado e relacionado ao curso de Estrutura de Dados 1 da Universidade de Brasília - Campus Gama, para cumprimento às exigências legais como requisito parcial à obtenção de parte da nota do semestre.

Orientador: Prof. Dr. Nilton Correia da Silva

BRASÍLIA - DF

JANEIRO/2023

RESUMO

Os diferentes algoritmos, formas de implementá-lo e flexibilidade da sua complexidade possibilitam cada vez mais a produção de programas completos e adequados para todos os tipos de máquina e funções. Um dos principais modelos de algoritmos utilizados em muitos casos são os algoritmos de ordenação, sendo diversificados e amplamente utilizados, seja para ordenar a média das notas de turmas ou até mesmo diversas linhas com quaisquer valores relacionados a um meio específico. Diante disso, foi proposto o trabalho de Busca e Ordenação de uma quantidade considerável de dados para a obtenção de algumas conclusões referentes ao uso do método de ordenação Bubble Sort.

LISTA DE FIGURAS

Figura 1 - Exemplo Blubble Sort.....	08
Figura 2 - Saída do terminal.....	08
Figura 3 - Especificações do Laptop.....	09

SUMÁRIO

1. INTRODUÇÃO	6
2. OBJETIVO	6
2.1 Objetivo Geral	6
2.2 Objetivo Específico	6
3. RELATO DAS ATIVIDADES	7
4. DESENVOLVIMENTO	7
5. RESULTADOS	8
6. ANEXOS	9

1. INTRODUÇÃO

O trabalho a ser apresentado trata-se de um relatório técnico da disciplina de Estrutura de Dados, ministrada pelo Prof. Dr. Nilton Correia da Silva sobre a atividade prática (Aplicação de Algoritmos de Busca e Ordenação em Registros de Operações de Vendas da Amazon.com). Serão tratadas informações acerca do proposto e do código produzido, bem como seu desenvolvimento o resultado obtido visando o proposto. Também será posto o funcionamento e complexidade algorítmica, que nos mostra como o software se comporta ao ser executado, medindo sua velocidade.

2. OBJETIVO

2.1 Objetivo Geral

O presente trabalho tem como objetivo relatar a atividade prática, sua realização e resultados obtidos, descrevendo o que foi produzido no que tange o proposto solicitado como resultado pelo orientador.

2.2 Objetivo Específico

- Realizar e Obter resultados a partir do trabalho em grupo
- Alocar Dinamicamente os dados vindos de um arquivo externo csv
- Implementar o algoritmo de ordenação Bubble Sort
- Gerar um arquivo de saída csv com os dados ordenados
- Indicar o tempo de execução e complexidade algorítmica

3. RELATO DAS ATIVIDADES

As atividades decorrentes do trabalho em grupo se deram por meio do Git e GitHub, que possibilitaram o trabalho em conjunto à distância dos membros. O desenvolvimento se deu a partir do programa VsCode, que é uma IDE (Ambiente de desenvolvimento integrado), possibilitando a programação utilizando a linguagem C.

O código foi concluído com êxito diante da atividade proposta, com a devida organização dos dados de entrada, gerando o arquivo de saída juntamente com as informações solicitadas no terminal, validando a pesquisa e mostrando os valores com ordenação por Bubble Sort.

4. DESENVOLVIMENTO

Colocado a proposta do trabalho, o grupo ficou com o desafio de abrir o arquivo csv, aloca-lo dinamicamente, ordená-lo com o algoritmo de ordenação Bubble Sort (Figura 1), que é um dos algoritmos de ordenação dos mais simples. Seu funcionamento se dá percorrendo o vetor diversas vezes, e a cada passagem, manda o maior elemento da sequência para o topo. Isso implica um grande tempo de processamento no pior caso, ou seja, quando há um desordenamento nos valores, demandando várias vezes a rodagem do trecho do algoritmo. Seu pior caso indica complexidade $O(n^2)$. No melhor caso, no qual o algoritmo já esteja ordenado, sua complexidade será $O(n)$.

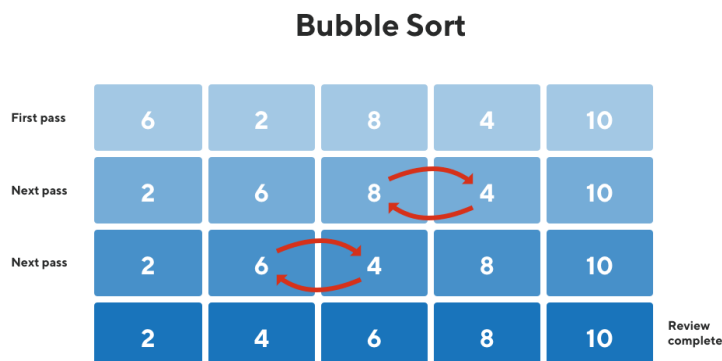


Figura 1 - Exemplo Bubble Sort.

Fonte: <[What is Bubble Sort? | Definition and Overview | ProductPlan](#)>, 2023.

5. RESULTADOS

Ao todo, todos os membros participaram ativamente auxiliando presencialmente ou pelas plataformas na confecção do código. Pelo proposto, foi obtido todos os resultados esperados com total funcionamento do código fonte. O tempo de execução (Figura 2) é advindo da execução do código em uma máquina com processador Intel Core I3 e 8 Gigabytes de Memória Ram (Figura 3), levando em conta que haviam outros programas sendo executados em primeiro e/ou segundo plano, também consumindo parte do processamento e memória.

```
Arquivo aberto com sucesso!  
Arquivo de saída gerado!  
A categoria com o maior valor: Set; 1281.000000  
A categoria com o menor valor: kurta; 0.000000  
O tempo de execucao foi de: 0.030000 segundos  
O metodo utilizado foi de Bubble sort que possui complexidade O(n^2)  
Sendo que para o melhor caso a complexidade diminui para O(n)
```

Figura 2 - Saída do terminal.

Fonte: Elaborada pelo autor, 2023.

Especificações do dispositivo	
Nome do dispositivo	LAPTOP-FNDCG7PO
Processador	Intel(R) Core(TM) i3-7020U CPU @ 2.30GHz 2.30 GHz
RAM instalada	8,00 GB (utilizável: 7,89 GB)
ID do dispositivo	CE4D557A-D148-4467-B1D7-D77C788DE548
ID do Produto	00327-30000-00000-AAOEM
Tipo de sistema	Sistema operacional de 64 bits, processador baseado em x64

Figura 3 - Especificações do Laptop.

Fonte: Elaborada pelo autor, 2023.

Também é possível observar outras informações e dados que foram solicitados para a saída, como mensagens de abertura e geração de arquivo, categoria de maior e menor valor, juntamente com o tempo de execução na máquina em segundos (Figura 3).

Com o arquivo original, contendo 128.974 linhas, executado pela mesma máquina, teve o tempo de execução em torno de 212 segundos.

6. ANEXOS

Código Fonte: https://github.com/NyndoND/EDA2022_2_GrupoH.git