

UNIVERSIDADE FEDERAL DA FRONTEIRA SUL



**GEX609 - Pesquisa e ordenação de dados (Ciência da Computação
(Turma B) - 3ª fase - Vespertino (Remoto) - 2020/1 - 2020/1)**

Trabalho 3 - Radix Sort

Any Cruz Silva

Matrícula: 1721101005

Chapecó

2020

1 - Enunciado

Implemente o algoritmo Radix Sort e repita com ele os testes de desempenho com os 3 tamanhos de vetor e os 3 tipos de ordenação inicial (crescente, decrescente e aleatória). Contabilize o tempo e faça as análises de forma análoga aos trabalhos 1 e 2.

2 – Considerações iniciais

Para realizar o presente trabalho foi utilizado um computador de seguinte configuração:

LG PC

Nome do dispositivo	Home
Processador	Intel(R) Pentium(R) CPU B980 @ 2.40GHz 2.40 GHz
RAM instalada	4,00 GB (utilizável: 3,88 GB)
ID do dispositivo	43CC63B2-D813-4116-8213- CC6A3636E0DD
ID do Produto	00327-60000-00000-AA229
Tipo de sistema	Sistema operacional de 64 bits, processador baseado em x64

O sistema operacional na ocasião apresentava o seguinte estado:

Especificações do Windows

Edição	Windows 10 Home Single Language
Versão	2004
Instalado em	22/08/2020
Compilação do SO	19041.508
Experiência	Windows Feature Experience Pack 120.2212.31.0

Em paralelo com a execução dos algoritmos estavam abertos os seguintes softwares: Visual Studio Code, Microsoft Office Word 2007, Prompt de comando do Windows (CMD) e o Explorador de arquivos.

Para executar os algoritmos foi utilizado CMD do Windows 10 com variável de ambiente configurada para realizar compilação. Para cada compilação o CMD foi fechado e aberto novamente para realizar a próxima execução, os resultados apresentados nas tabelas são resultados de médias de tempo entre 3 execuções de cada número de constantes para cada método e ordenação.

3 – Radix Sort com ordenação em ordem crescente

Primeiramente foi feita a execução do Radix Sort com 10.000, 50.000 e 100.000 constantes geradas aleatoriamente foram obtidos os seguintes resultados;

Número de constantes	Tempo
100.000	28 segundos
50. 000	19 segundos
10.000	11 segundos

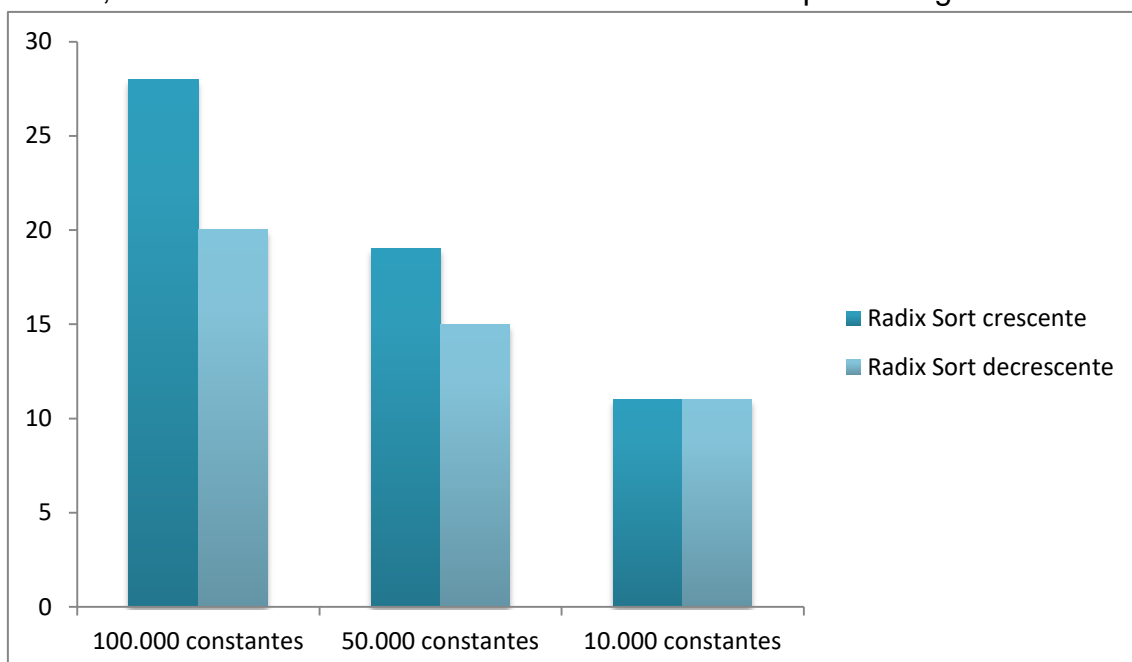
3.1 – Radix Sort com ordenação em ordem decrescente

Primeiramente foi feita a execução do Radix Sort com 10.000, 50.000 e 100.000 constantes geradas aleatoriamente foram obtidos os seguintes resultados;

Número de constantes	Tempo
100.000	20 segundos
50. 000	15 segundos
10.000	11 segundos

4 – Gráficos comparativos

4.1 – Gráfico comparativo de tempo levado por cada algoritmo para ordenar 100.00, 50.000 e 10.000 constantes no decorrer do tempo em segundos:



5 – Conclusões

O Radix Sort é um método de ordenação que tem como principal característica não utilizar comparações entre as chaves de um conjunto e sim o processamento da chave por partes. Para realizar a ordenação o método divide a chave em várias partes, existem duas modalidades para esse processo que é a ordenação considerando o dígito menos significativo (LSD, less significant digit) e a ordenação considerando o dígito mais significativo (MSD, more significant digit).

No presente trabalho foram realizados testes utilizando a versão do algoritmo que inicia a ordenação considerando o dígito menos significativo de uma chave (LSD). Também foram realizados testes com o algoritmo em situações onde a ordenação é feita em ordem crescente e decrescente, para preencher o vetor foram utilizados valores gerados de forma aleatória.

A complexidade do algoritmo em seu pior caso é $O(n)$, apresenta as vantagens de ser um método estável e não comparativo.