



Pesquisa e Ordenação de Dados

(Ordenação de Dados - Heap Sort, Shell Sort, Quick Sort)

Prof. Dr. Joel da Silva

joel.silva@ufsm.br

www.joeldasilva.com.br



UFSM

Universidade Federal de Santa Maria

Departamento de Tecnologia da Informação

Shell Sort

- Definição
 - Criado por **Donald Shell**. É um **refinamento** do método de **inserção direta**. Difere do método de inserção direta pelo fato de no lugar de considerar o array a ser ordenado como um único segmento, ele considera vários segmentos sendo aplicado o método de inserção direta em cada um deles. O algoritmo **passa várias vezes pela lista dividindo o grupo maior em menores**. Nos grupos menores é aplicado o método da ordenação por inserção.
- Simples de Ser implementado
- Um dos mais rápidos dentre os algoritmos de complexidade quadrática
- In-place
 - Não necessita de vetor auxiliar

Shell Sort

- Não estável
 - Elementos iguais são trocados de posição
- Adaptativo: dependendo da entrada, pode ser mais rápido
- Complexidade
 - No melhor caso será $O(n \log n)$ e no pior será $O(n^2)$
- Material para estudo: <https://goo.gl/MFo8fv>

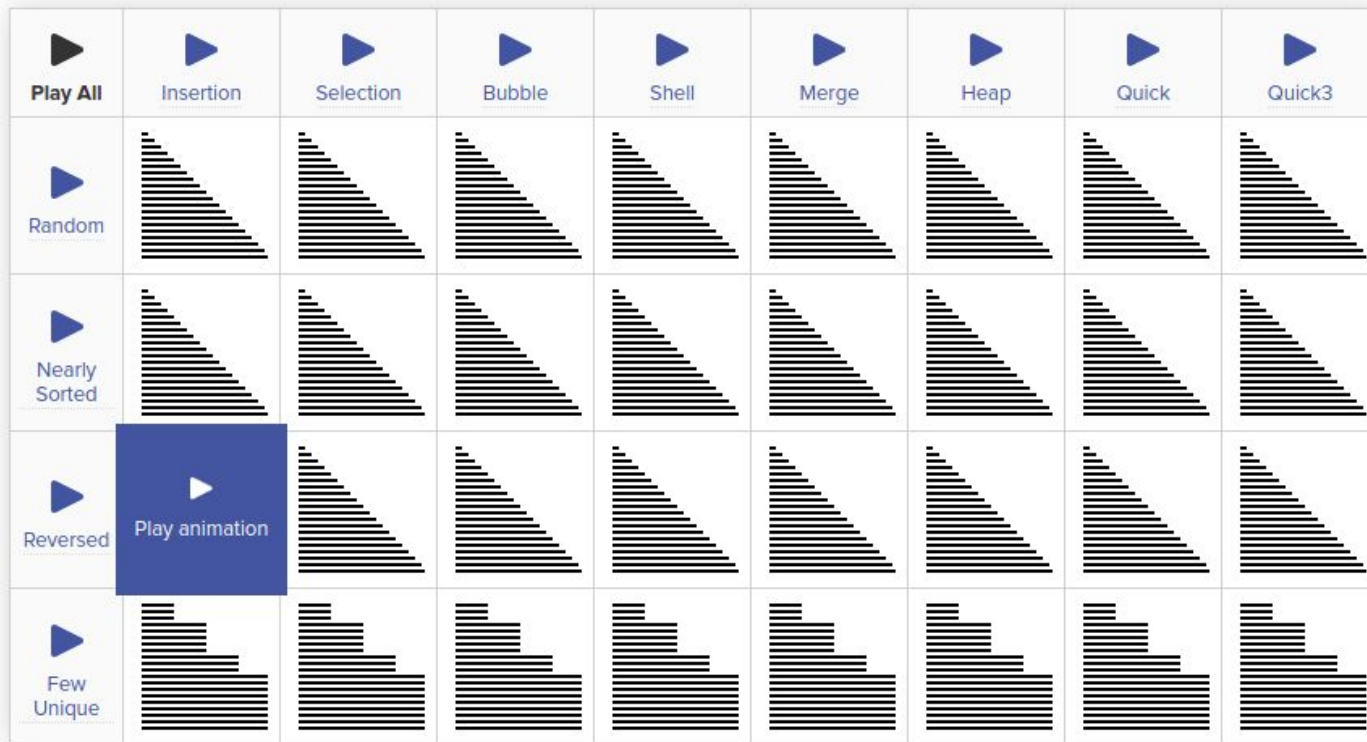
Shell Sort

- Demonstrando seu funcionamento
 - <https://www.toptal.com/developers/sorting-algorithms/shell-sort>



Shell Sort

- Comparando com outros algoritmos
 - <https://www.toptal.com/developers/sorting-algorithms>



Quick Sort

- Definição
 - Inventado por Hoare em 1960.
 - Adota a estratégia de **divisão e conquista**. A estratégia consiste em rearranjar as chaves de modo que as chaves "menores" precedam as chaves "maiores". Em seguida o *quicksort* ordena as duas sublistas de chaves menores e maiores recursivamente até que a lista completa se encontre ordenada.
- Simples de Ser implementado
- Um dos mais rápidos dentre os algoritmos de complexidade quadrática
- In-place
 - Não necessita de vetor auxiliar

Quick Sort

- Não estável
 - Elementos iguais são trocados de posição
- Adaptativo: dependendo da entrada, pode ser mais rápido
- Complexidade
 - No melhor caso será $O(n \log n)$ e no pior será $O(n^2)$
- Material para estudo: <https://goo.gl/MFo8fv>

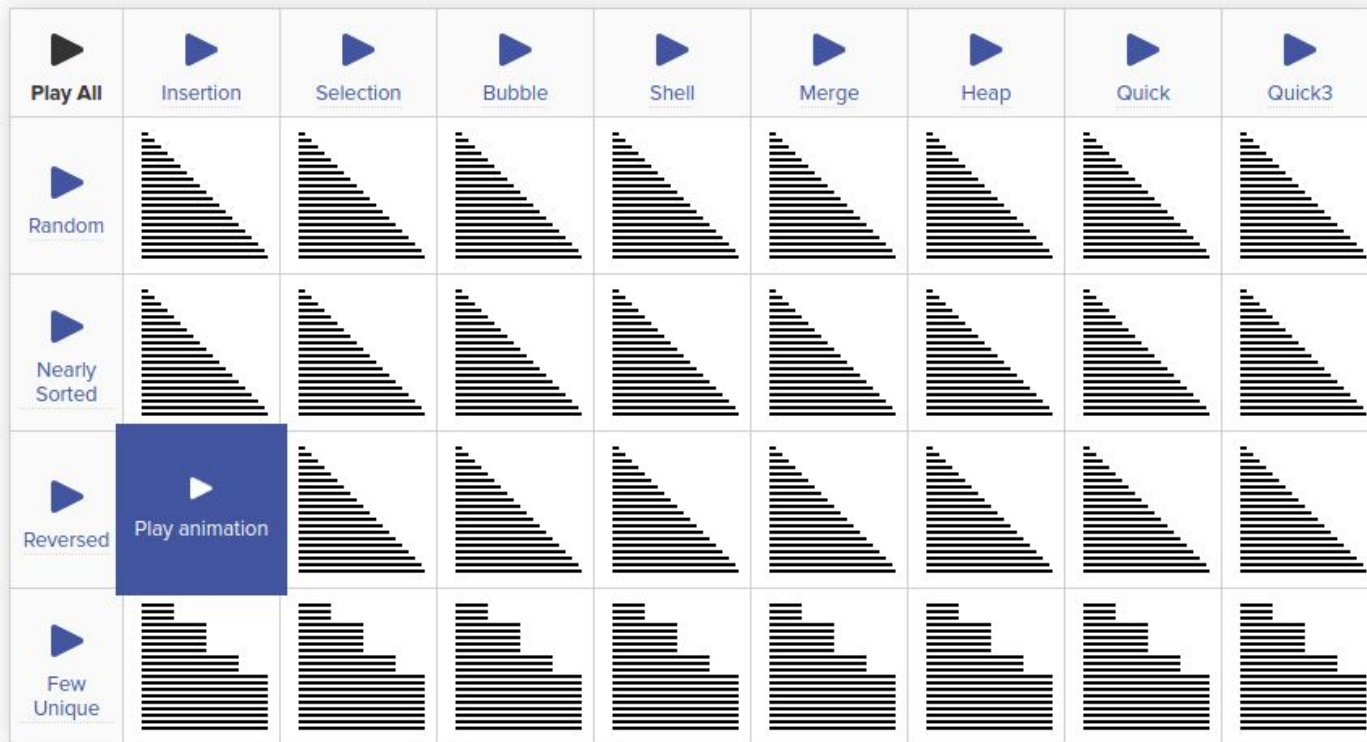
Quick Sort

- Demonstrando seu funcionamento
 - <https://www.toptal.com/developers/sorting-algorithms/quick-sort>



Quick Sort

- Comparando com outros algoritmos
 - <https://www.toptal.com/developers/sorting-algorithms>



Heap Sort

- Definição
 - É um algoritmo de ordenação generalista, e faz parte da família de algoritmos de ordenação por seleção. Foi desenvolvido em 1964 por Robert W. Floyd e J.W.J Williams.
 - Utiliza uma estrutura de dados chamada **Heap** para ordenar os elementos a medida que os insere na estrutura. Assim, ao final das inserções, os elementos podem ser sucessivamente removidos da raiz da heap, na ordem desejada, sendo essencial que a propriedade **max-heap** seja mantida. Essa propriedade garante que o valor de todos os nós são menores que os de seus respectivos pais.

Heap Sort

- In-place
 - Não necessita de vetor auxiliar
- Não Estável
 - Elementos iguais são trocados de posição
- Não Adaptativo: Independe da entrada
- Complexidade
 - $O(n \log n)$
- Material para estudo: <https://goo.gl/MFoé8fv>

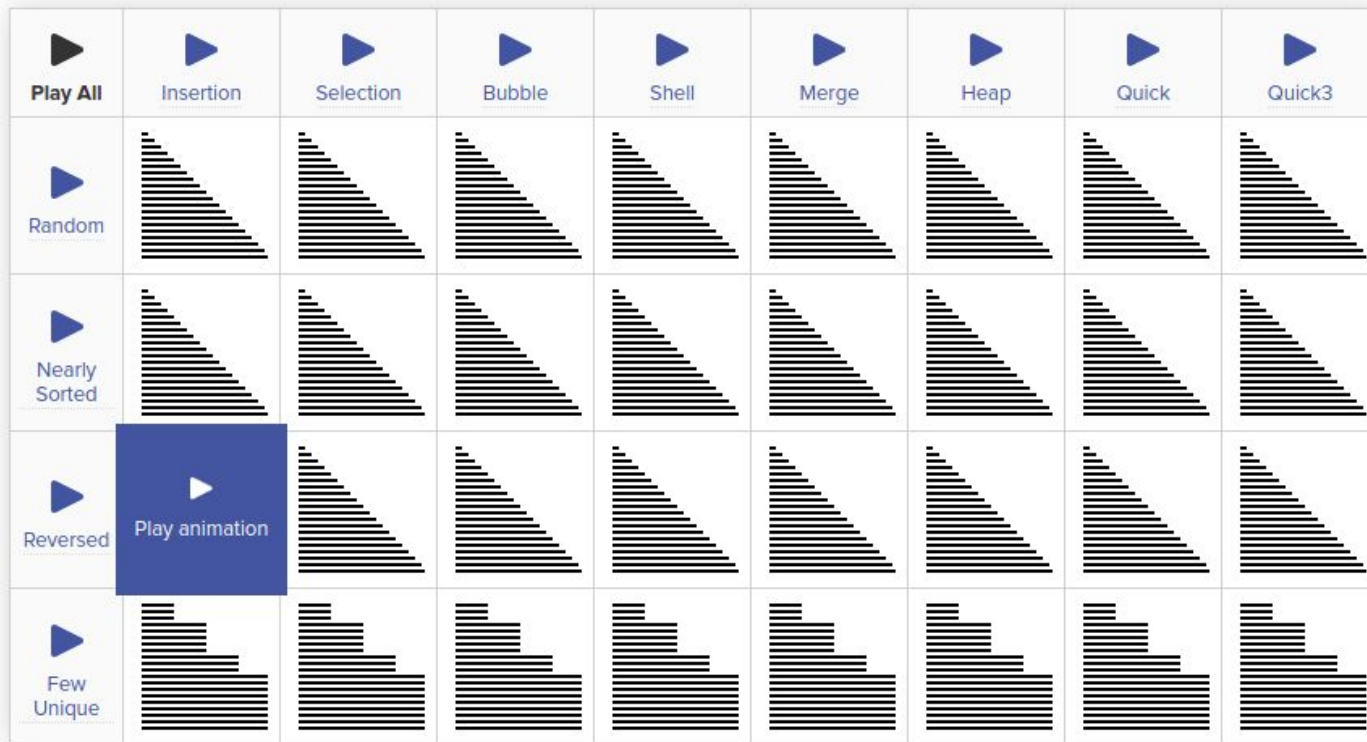
Heap Sort

- Demonstrando seu funcionamento
 - <https://www.toptal.com/developers/sorting-algorithms/heap-sort>



Heap Sort

- Comparando com outros algoritmos
 - <https://www.toptal.com/developers/sorting-algorithms>



Material para Estudo

- Mapa Mental
 - <https://goo.gl/CWeqZ3>

Material para Estudo

Exercícios

- Implementar e testar os algoritmos ShellSort, HeapSort, QuickSort na Classe OrdenaInteirosApp.java, disponibilizada no GitLab;
 - A atividade deverá ser entregue até a próxima aula fará parte da avaliação semestral (1 ponto);
- Implementar e testar os algoritmos ShellSort, HeapSort, QuickSort na Classe Contatos.java, disponibilizada no GitLab, para ordenar por idade e/ou nome;
 - A atividade deverá ser entregue até a próxima aula e fará parte da avaliação semestral (1 ponto);