# Manual utilizador

Índice

[Manual utilizador 1](#_Toc353557965)

[Insertion-Sort 3](#_Toc353557966)

[Bubble-Sort 3](#_Toc353557967)

[Arquitetura do código 3](#_Toc353557968)

# Insertion-Sort

A ordenação por Insertion-Sort seleciona o maior ou o menor elemento e coloca-o em uma posição correta dentro do array em ordenação.

Durante a troca o array é transformado em duas listas, uma com os elementos ordenados e outra com os elementos não ordenados. No inicio da execução a lista ordenada esta vazia, e a lista desornada cotem todos os elementos desordenados, quando o processo termina a lista ordenada apresenta(n-1)elementos e a desordenada, terá apenas um elemento, ou seja, inicialmente é encontrado o maior elemento e depois faz as comparações com os outros. Consequentemente , cria-se uma sublista desordenada(n-2) elementos e uma sublista ordenada com 2 elementos de seguida o processo repete-se até que a lista tenha (n-1).

<http://upload.wikimedia.org/wikipedia/commons/0/0f/Insertion-sort-example-300px.gif>

# Bubble-Sort

A ordenação por Bubble-Sort consiste em analisar sequencialmente cada elemento do array comparando os elementos vizinhos entre si, ou seja, o objetivo é percorrer o array várias vezes pelo elemento maior da sequencia, comparando os pares de elementos da esquerda para a direita.

# Merge Sort

Ao contrario dos algoritmos a cima o Merge Sort contem duas listas sendo uma delas com elementos já ordenados em seguida o Merge Sort quebra as listas até que elas tenham esse tamanho único e depois as une, recursivamente.

Este algoritmo baseia-se na conquista e combinação.