

## 1 Objectivos

Pretende-se a implementação de 2 algoritmos de ordenação. Devem os alunos apresentar a implementação de 2 algoritmos de ordenação, que recebam um array de objectos Comparable e ordenem o array. Não devem os alunos sem autorização do professor fazer uso de classes do package "java.util", excepto para input/output.

## 2 O Trabalho

O trabalho consiste na implementação de 2 algoritmos de ordenação distintos e de complexidade distinta. Os algoritmos escolhidos poderão ser algum(s), dos algoritmos dados/estudados ou não. O trabalho será classificado tendo em conta os algoritmos escolhidos e o grau de dificuldade dos mesmos. Não podem, sob qualquer pretexto, fazerem os alunos copy/paste, de quaisquer fontes onde se apresentem implementações de algoritmos, devendo os alunos estudar as referidas fontes e realizar a sua própria implementação. É o objectivo deste trabalho prático. A implementação apresentada, deve receber do standard input uma sequência de valores (comparáveis) e realizar a sua ordenação. Deverá estruturar o seu trabalho por forma a que o ficheiro pedido contenha obrigatoriamente, os seguintes métodos:

- um método estático da forma **public static void method1(Comparable[] A)** que implementa a ordenação do array A, segundo o método **method1**
- um método estático da forma **public static void method2(Comparable[] A)** que implementa a ordenação do array A, segundo o método **method2**
- Um método **public static Comparable[] le\_array()** que lê o array de valores do standard input.
- Um método **public static void printArray(Object[] x)** que lista para o standard output, o array passado por argumento, na forma "tudo na mesma linha, separado por ; muda de linha quando acaba".
- Um **main** que executa a leitura, ordenação e listagem do array ordenado, usando os métodos implementados.

Funcionalidade	Interface	Implementação
Leitura de dados e respectiva ordenação	-	Sort.java

Table 1: Classes a implementar

## 2.1 Entrega

O trabalho pode ser realizado individualmente ou em grupos de 2 alunos. Os alunos devem manter os mesmos grupos que formaram para a realização do(s) trabalho(s) anteriores. A data limite para a submissão do trabalho é dia 22 de Janeiro de 2019, sendo realizada a submissão pelo moodle, nos moldes habituais. Deverão submeter adicionalmente ao ficheiro Sort.java um ficheiro pdf com um relatório do trabalho realizado. O relatório deverá explicar/apresentar os algoritmos que o grupo implementou, detalhando o mais possível a apresentação do algoritmo(não a sua implementação). Não encham o relatório com detalhes de implementação. A nota dada ao trabalho, está relacionada também com o relatório. Os relatórios devem identificar os alunos que constituem o grupo, bem como no início do ficheiro java devem ser identificados os alunos que realizaram o trabalho. Não há desistências: trabalhos/relatórios recebidos e identificados, serão avaliados de igual modo para todos os membros do grupo, que seja 5 quer seja 0. Trabalhos em que seja detectada fraude( no código, no relatório ou na identificação dos membros do grupo) serão anulados. Os trabalhos não serão apresentados oralmente, a menos que exista alguma irregularidade. Todos os ficheiros deverão ser "zipados" e submetidos num único ficheiro com o número dos alunos que realizam o trabalho.