 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Trabalho Prático	Ano letivo 2017/2018	Data
	Licenciatura em Engenharia Informática		Hora
	Estruturas de Dados		Duração

Observações:

V 1.0 – Versão inicial

## A. Destinatários

Este projeto destina-se a todos os estudantes inscritos na Unidade Curricular de Estruturas de Dados, da Licenciatura em Engenharia Informática.

O projeto tem um peso de 60% na classificação final da UC e tem como requisito mínimo uma classificação de 9.5 valores.

## B. Objetivos

Este projeto funcionará como um elemento integrador dos conhecimentos adquiridos nas duas tipologias da UC de Estruturas de Dados.

Os objetivos específicos são os seguintes:


- Compreender e dominar os conhecimentos teóricos e práticos sobre Estruturas de Dados e os algoritmos que lhes estão associados;
- Saber escolher, de forma fundamentada e para cada problema específico, qual a estrutura de dados abstrata e implementação mais adequada;
- Implementar algoritmos escaláveis sobre estruturas de dados potencialmente grandes;
- Adquirir competências com vista à resolução de problemas compostos;
- Estimular o trabalho em equipa como elemento essencial do processo de aprendizagem individual.

## C. Realização

O trabalho é realizado em grupos de 2 elementos, salvo situações específicas a validar previamente com o regente da UC.

O trabalho deve ser realizado em linguagem JAVA. Não é permitida a utilização de qualquer membro da API de Collections do JAVA. É permitida a utilização da classe `java.util.Iterator`.

Juntamente com este enunciado é fornecida uma biblioteca que contém, entre outros elementos, as interfaces de todas as coleções necessárias para a realização do trabalho, similares às interfaces utilizadas em contexto de aula. A utilização destas interfaces na realização deste trabalho é de carácter obrigatório. Cada grupo deve utilizar as implementações destas interfaces desenvolvidas pelos seus elementos em contexto de aula.

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Trabalho Prático	Ano letivo 2017/2018	Data
	Licenciatura em Engenharia Informática		Hora
	Estruturas de Dados		Duração

## D. Relatório

A submissão do trabalho deve ser acompanhada de um relatório escrito. Este relatório deve abordar claramente para além de outros elementos que o grupo considere importantes, os seguintes tópicos:

- Incluir um diagrama de classes que ilustre a estrutura do projeto desenvolvido;
- Fundamentação dos tipos de dados abstratos escolhidos para cada problema específico;
- Fundamentação das implementações utilizadas para cada tipo de dados abstratos (e.g. ArrayList vs. LinkedList);
- O conjunto de funcionalidades implementadas;
- Descrição dos algoritmos implementados considerados mais complexos/relevantes (e.g. pesquisa do caminho mais curto);
- Descrição de testes efetuados e seu resultado (e.g. para um determinado problema P, a função F com o input I deu o resultado R).

## E. Realização e Submissão

Este trabalho é realizado em grupos de dois elementos da mesma turma, salvo situações pontuais a validar previamente com os docentes da UC.

O projeto é implementado no IDE Netbeans. Qualquer ferramenta adicional utilizada deve ser mencionada no relatório.

Ambos os elementos do grupo submetem o trabalho desenvolvido.

A submissão deverá consistir num ficheiro .zip cujo nome deve seguir a norma ED\_NumAluno (e.g. ED\_80000) com os seguintes conteúdos:

- Grupo.txt – ficheiro contendo informação (nome e número) dos elementos do grupo
- Projeto – pasta contendo o projeto Netbeans com a implementação do trabalho;
- Relatório.pdf – ficheiro .pdf contendo o relatório do trabalho desenvolvido


Os elementos a avaliar deverão ser submetidos até às 23:55 do dia 26 de Janeiro de 2018, usando para tal a página da unidade curricular de ED em <http://www.estg.ipp.pt/moodle>, dentro do prazo previsto.

Cada grupo procederá, obrigatoriamente, à apresentação e defesa do seu trabalho na data e local estipulados no calendário de exames do presente ano letivo. A defesa será individual, do que decorre que elementos de um mesmo grupo podem ter avaliações diferentes.

## F. Critérios de Avaliação

A nota de cada estudante será determinada por três componentes principais:

- A. Desempenho individual durante a defesa
- B. Qualidade do projeto (e.g. escolha de tipos de dados abstratos, escolha de implementações)
- C. Performance/correção do projeto (e.g. resultado dos testes efetuados às funções)
- D. Qualidade do relatório

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Trabalho Prático	Ano letivo 2017/2018	Data
	Licenciatura em Engenharia Informática		Hora
	Estruturas de Dados		Duração

Para avaliar a componente A, será pedido a cada estudante, durante a defesa:

- A correção de funcionalidades eventualmente mal implementadas;
- A implementação de novas funcionalidades;
- A reimplementação de funcionalidades já existentes.


As questões a colocar poderão incluir qualquer elemento (incluindo elementos teóricos) da matéria lecionada na UC. Serão avaliados o desempenho do aluno e o seu conhecimento dos conteúdos abordados na UC.

## G. Enunciado

Um formigueiro é a habitação de formigas, seres altamente sociais e pertencentes à família mais numerosa de insetos do mundo. Cada formigueiro é constituído por várias salas, ligadas entre si por túneis. Em determinadas espécies, cada colónia (ou conjuntos interligados de colónias) pode incluir milhares destas salas, que se espalham ao longo de vários quilómetros quadrados. Parte do sucesso destas sociedades de insetos assenta na sua inteligência coletiva, sendo capazes de resolver em conjunto problemas altamente complexos. O objetivo deste trabalho é simular, ainda que de forma simplificada, o funcionamento de um formigueiro, implementando alguns dos problemas aí existentes.

Serão modelados os seguintes conceitos:

- Comida – Cada comida tem um identificador único e um tamanho. Uma comida com tamanho  $> n$  pode ser partida em  $n$  comidas de tamanho 1 numa sala de Processamento (ver abaixo)
- Formiga – Uma formiga é caracterizada por um identificador único
  - Uma formiga pode carregar comida entre diferentes salas, tendo, portanto, também uma capacidade de carga máxima (número de comidas máximo que consegue transportar)
  - A capacidade de carga máxima de transporte não tem qualquer relação com o tamanho da comida, i.e., uma formiga com capacidade máxima de 3 consegue transportar simultaneamente no máximo 3 comidas, independentemente do tamanho de cada comida
- Sala – Um formigueiro é constituído por 1 ou mais salas. Cada sala é caracterizada por um identificador numérico único, uma descrição, e as suas coordenadas X e Y no plano bidimensional. Para além disso, cada sala pode ter, a cada momento, zero ou mais formigas presentes (assuma que não existe um limite máximo para o número de formigas presentes numa sala num dado momento). Existem ainda dois tipos de sala mais específicos:
  - Silo – Um silo é utilizado pelas formigas para guardar comida. O silo é uma estrutura vertical em que as formigas depositam comida uma em cima da outra. Quando as formigas retiram comida, é sempre a comida que se encontra no topo do silo que é retirada. Assuma que o silo não tem limite de espaço
  - Sala de Processamento – A sala de processamento é utilizada pelas formigas para processar (partir) a comida, funcionando como uma linha de produção. Ou seja, a nova comida que é depositada nesta sala é depositada no fim da linha de produção. Também aqui pode assumir que não existe um limite de espaço. Quando uma formiga pretende retirar comida desta sala, uma de duas coisas pode acontecer:

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Trabalho Prático	Ano letivo 2017/2018	Data
	Licenciatura em Engenharia Informática		Hora
	Estruturas de Dados		Duração

- A próxima comida a retirar tem tamanho 1, sendo removida da sala de processamento e carregada na formiga
- A próxima comida a retirar tem tamanho  $n > 1$ , sendo acrescentadas  $n$  comidas de tamanho 1 ao fim da linha de produção da sala; Nesta situação, a formiga não é carregada com qualquer comida
- Formigueiro – O formigueiro contém no seu interior uma ou mais salas, bem como formigas (sempre localizadas numa sala) e comida (que em cada momento pode estar numa formiga, num silo ou numa sala de processamento). Para além disso:
  - Um formigueiro tem sempre uma sala que é identificada como sendo a entrada do formigueiro
  - Entre cada duas salas pode existir, no máximo, um túnel que permite o trânsito de formigas entre essas duas salas em ambos os sentidos
  - Cada túnel é caracterizado por um identificador único, um comprimento (que é calculado em função da distância no plano bidimensional entre as duas salas que liga) e um raio (com o valor mínimo de 1)
  - Uma formiga que não transporta comida pode circular em qualquer túnel
  - Uma formiga que transporta  $n$  comidas (independentemente do tamanho de cada comida) apenas pode circular em túneis cujo raio seja maior ou igual a  $n$

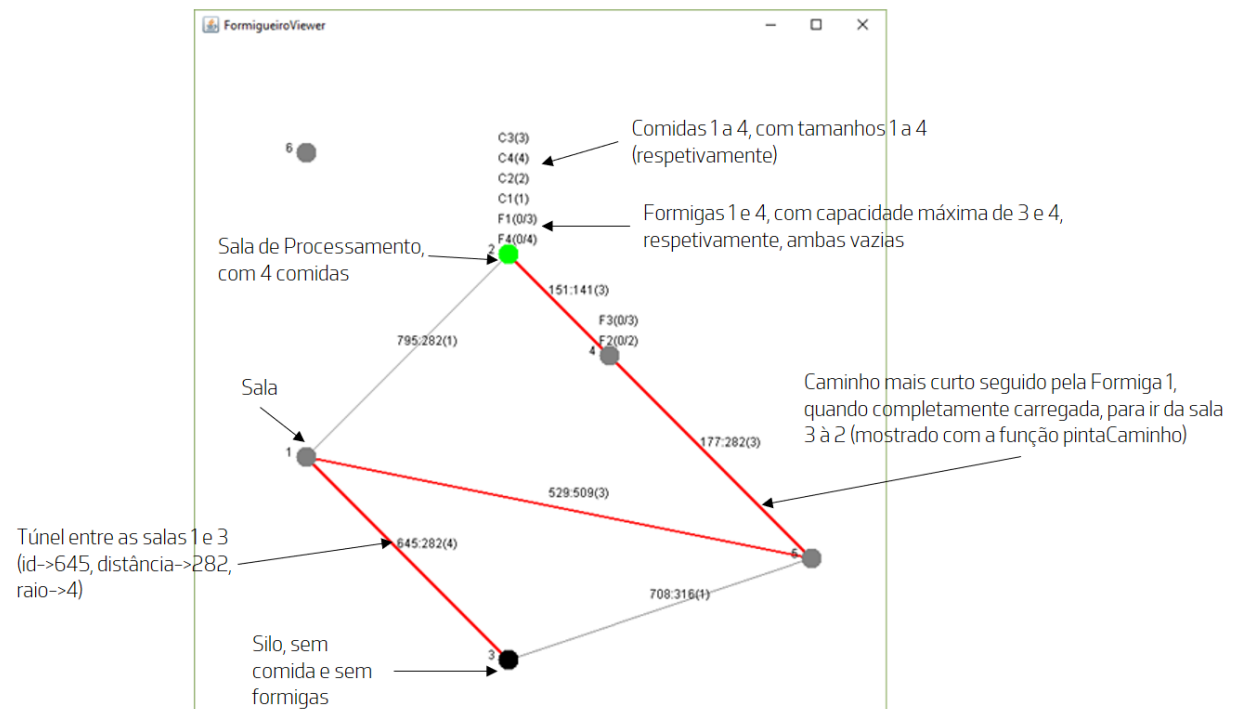
## H. Recursos

Juntamente com este enunciado é fornecida uma biblioteca (FormigueiroRecursos.jar) que contém os seguintes packages e respetivo conteúdo:

- `recursos.exceptions` – Contém as exceções a utilizar na implementação do trabalho
- `recursos.interfaces` – Contém os interfaces que regem a implementação dos elementos do formigueiro (e.g. IFormiga, ISala, etc.)
- `recursos.interfaces.collections` – Contém os interfaces dos tipos abstratos de dados a utilizar obrigatoriamente na resolução deste trabalho. Dependendo da implementação, alguns tipos abstratos poderão não ser necessários
- `recursos.utils` – Contém a classe FormigueiroViewer que permite visualizar graficamente instâncias de IFormigueiro. Mais detalhes sobre a utilização desta classe estão disponíveis na sua documentação. A Figura 1 mostra um exemplo da visualização de um IFormigueiro com esta classe, em que se pintou um caminho.

É ainda fornecida a documentação completa da biblioteca, em alguns casos ilustrada com exemplos do resultado esperado de um método para um determinado input.

<div>P.PORTO</div> <div>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</div>	Trabalho Prático	Ano letivo 2017/2018	Data
	Licenciatura em Engenharia Informática		Hora
	Estruturas de Dados		Duração



**Figura 1:** Exemplo da utilização da classe `recursos.utils.Viewer` para um determinado formigueiro, em que foi desenhado o resultado da invocação de uma função que acha o caminho mais curto entre dois pontos, para uma formiga com uma determinada carga.