 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame de Época Especial	Ano letivo 2017/2018
	Curso LEI – Licenciatura em Engenharia Informática LSIRC - Licenciatura em Segurança Informática em Redes de Computadores	
	Unidade Curricular Paradigmas da Programação	

## Destinatários

Este trabalho destina-se a todos os estudantes inscritos na unidade curricular de Paradigmas da Programação que pretendam obter aprovação à Unidade Curricular.

## Objetivos

Com a realização do trabalho prático, pretende-se que os alunos ponham em prática todos os conhecimentos adquiridos na utilização do paradigma de programação orientado a objetos e a sua implementação na linguagem de programação Java, demonstrando as suas apetências em:

- Conhecer e compreender os conceitos fundamentais associados ao paradigma da programação orientada a objetos;
- Conceber e implementar, para problemas concretos, soluções que tenham por base o paradigma da programação orientada a objetos.
- Reconhecer e compreender a semântica e a sintaxe da linguagem Java.
- Reutilizar, alterar e desenvolver código recorrendo à linguagem Java tendo em vista um determinado problema com regras semânticas específicas.

Considere ainda que:

- Não é permitida a utilização de API's/conceitos Java que não tenham sido alvo de lecionação - ano letivo 2017-2018 - da unidade curricular Paradigmas de Programação. Os alunos que pretendam utilizar API's adicionais devem atempadamente pedir autorização a um dos docentes da unidade curricular.
- Não é permitida a utilização de coleções Java predefinidas (*Java Collections Framework*).
- Os recursos de suporte ao trabalho referenciados no enunciado, são de utilização **obrigatória**.

## Enunciado

A empresa de produção de videojogos *ARGA* (Advanced Research Game Agency), decidiu organizar uma competição de programação para captação de novos talentos para a empresa. Neste contexto, a empresa requer a construção de um jogo de arcade clássico parecido com o famoso jogo Tetris. Para o desenvolvimento deste jogo, a empresa possui uma serie de recurso que disponibiliza aos potenciais candidatos. Para ilustração, o jogo a desenvolver deverá ter o aspeto presente na figura 1.

O objetivo é avaliar a capacidade de desenvolvimento dos potenciais talentos da do Vale do Sousa de forma a assegurar as melhores contratações.

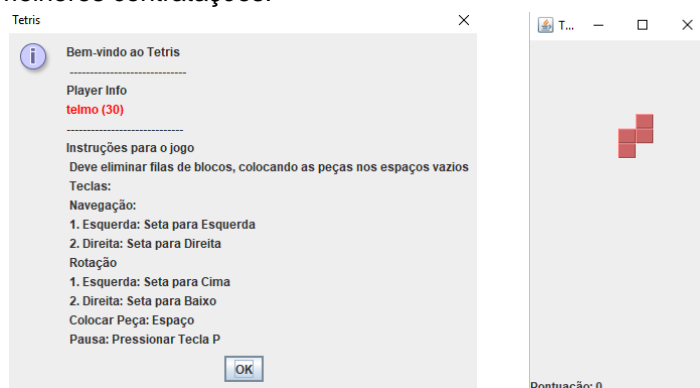



Figura 1 - Jogo Clássico a desenvolver

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame de Época Especial	Ano letivo 2017/2018
	Curso LEI – Licenciatura em Engenharia Informática LSIRC - Licenciatura em Segurança Informática em Redes de Computadores	
	Unidade Curricular Paradigmas da Programação	

Os requisitos estabelecidos pela ARGA são os seguintes:

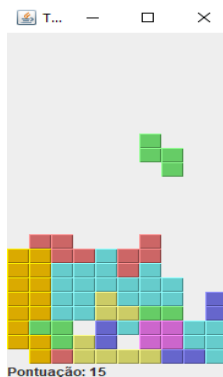
- O jogo deverá ser desenvolvido em linguagem Java e utilizando os recursos disponibilizados (disponíveis na plataforma moodle);
- Deverá ser possível controlar apenas o bloco em jogo podendo mudar de posição e rodar a forma;
- Deverá ser possível a visualização da pontuação obtida no final de cada jogo;
- A velocidade de descida dos blocos deve aumentar com o passar do tempo;
- Após uma linha completa na horizontal da grelha de jogo deve ser dada uma pontuação extra e os respetivos blocos desaparecerem do jogo deixando os blocos acima descer uma posição vertical na grelha;

Para além das APIs de suporte, a ARGA disponibiliza uma biblioteca com os contratos base (2018.AE.GameResources) que devem (no mínimo) ser implementados para a realização deste trabalho.

```
Player p = new Player("telmo", 30);
Board game = new Game(p, "file.txt");
game.start();
```


*Figura 2 - Início de uma sessão de jogo*

Para iniciar o jogo, os programadores devem no mínimo invocar no método main as instruções necessárias de forma a criar um a grelha de jogo (Figura 2).



*Figura 3- Sessão de jogo*

Na Figura 3, podemos ver o uma sessão do jogo clássico onde são mostrados o jogo a decorrer, pontuação do jogo, formas, e a forma em jogo. Após o início do jogo as formas devem aparecer aleatoriamente e o jogador deve ir encaixando da melhor forma as peças no fundo da grelha. A informação sobre a pontuação do jogador é gerida automaticamente desde que se encontrem implementadas todas as funcionalidades presente no ficheiro de recursos.

 ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO	Tipo de Prova Exame de Época Especial	Ano letivo 2017/2018
	Curso LEI – Licenciatura em Engenharia Informática LSIRC - Licenciatura em Segurança Informática em Redes de Computadores	
	Unidade Curricular Paradigmas da Programação	

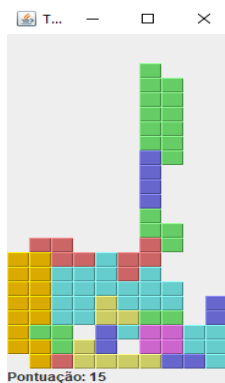


Figura 4 - Fim de jogo

Na Figura 4 é mostrado um dos possíveis desfechos do jogo com um resumo do jogo com a pontuação final do jogador.

### Requisitos Mínimos


Numa primeira fase, torna-se necessária a criação de uma API em linguagem Java que permita expor as funcionalidades inerentes dos vários elementos que constituem o jogo. Deverá incluir pelo menos as seguintes funcionalidades:

- O relacionamento entre os diversos conceitos encontra-se descrito nos recursos disponibilizados para a implementação da API e no javadoc. A leitura e interpretação e implementação das funcionalidades presentes no javadoc presente no ficheiro de recursos é obrigatória;
- Deverá ser possível jogar com as diferentes formas, movimenta-las na direção horizontal e rodar a peça sobre si enquanto a peça se encontra em jogo.
- Deverá desenvolver classes que implementem todas as classes e métodos abstratos e interfaces presentes no ficheiro de recursos bem como proceder de forma a que o jogo possa iniciado e jogado;
- Deverá ser capaz de mostrar a classificação do jogador;
- Deverá armazenar num ficheiro todas as pontuações obtidas por todos os jogadores na plataforma;
- Deverá carregar de um ficheiro todas as pontuações obtidas por todos os jogadores na plataforma, num ranking ordenado.

### Requisitos Bonificação


Tendo em consideração que a biblioteca de recursos disponibilizada para o desenvolvimento do jogo apenas permite visualizar uma sessão de jogo, deve considerar desenvolvimentos adicionais no trabalho para atingir os critérios de bonificação. Assim sendo deverá ter em consideração as seguintes funcionalidades:

- Elaborar um pré menu no início do jogo capaz de capturar o nome do jogador e apresente as restantes funcionalidades de bonificação presentes;
- Procurar a pontuação de um jogador na lista de pontuação a partir do seu nome;
- Dada uma pontuação verificar qual é a pontuação do jogador imediatamente a seguir no ranking;
- Mostrar a média de pontos de todos os jogadores;
- Armazenamento de recordes pessoais de cada jogador em ficheiros;

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame de Época Especial	Ano letivo 2017/2018
	Curso LEI – Licenciatura em Engenharia Informática LSIRC - Licenciatura em Segurança Informática em Redes de Computadores	
	Unidade Curricular Paradigmas da Programação	

- Dado um nome do jogador indicar quantas partidas já foram efetuadas pelo jogador, a pontuação mais alta, a pontuação mais baixa e a soma do total de pontos conquistados nas diversas partidas.

Outras bonificações sugeridas pelos alunos poderão ser aceites mediante validação pelos docentes da disciplina no momento da apresentação do trabalho. É encorajado o desenvolvimento de funcionalidades que façam sentido no contexto do jogo e melhorem a experiência de jogo.

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame de Época Especial	Ano letivo 2017/2018
	Curso LEI – Licenciatura em Engenharia Informática LSIRC - Licenciatura em Segurança Informática em Redes de Computadores	
	Unidade Curricular Paradigmas da Programação	

## Elaboração do trabalho

Este trabalho é realizado individualmente por alunos inscritos em época especial na unidade curricular de Paradigmas de Programação.

## Datas e considerações

O trabalho deve ser entregue até às **23h55** horas do dia **11 de Setembro de 2018**, devendo a entrega ser feita através da página da unidade curricular de Paradigmas de Programação em <http://moodle.estgf.ipp.pt>. Será ainda realizada a defesa do trabalho no dia **12 de Setembro de 2018**

Considera-se por **defesa satisfatória**, quando o aluno **prova** que realizou o trabalho submetido e que **domina todos** os conceitos de programação orientada a objetos aplicados na resolução do trabalho. Tentativas de **fraude**, resultarão na avaliação do trabalho como: **Fraude Académica**.

## Observações:

A entrega dos trabalhos deve obedecer aos requisitos identificados no ponto **Formato da entrega**. Os alunos que não realizem a entrega do trabalho até à data/hora definida serão sujeitos a penalização.

## Formato da entrega

Os trabalhos entregues deverão evitar (se possível) utilizar caminhos absolutos ou endereços específicos, de modo a que possam ser facilmente utilizados em qualquer máquina. Para além disso, e no sentido de facilitar a receção dos vários trabalhos recebidos, estes deverão observar as seguintes regras:

- **Todos os elementos do grupo** deverão submeter o trabalho no link respetivo (**Entrega do Trabalho**);
- O trabalho desenvolvido deverá ser entregue através do moodle, através da submissão de um ficheiro com o nome PP\_AE\_<nr\_do\_aluno>.zip, contendo:
  - Os ficheiros criados incluindo o(s) projeto(s) do IDE Netbeans e uma pasta com a distribuição (jar) da solução proposta.
  - Recorra a comentários JavaDoc, e não só, de modo a documentar, o mais exaustivamente possível, o código desenvolvido.
  - Cada ficheiro de código entregue por cada grupo terá de possuir no início do mesmo um comentário com pelo menos a seguinte informação (com as adaptações óbvias para cada aluno/grupo):

```

/*
 * Nome: <Nome completo do aluno>
 * Número: <Número mecanográfico do aluno>
 * Turma: <Turma do aluno>
 *
 */

```