



# Padrões e Desenho de Software

2025/26

## 2º Trabalho Prático

**Número de alunos por grupo: 2**

**Data limite de entrega (estudantes normais): 9-1-2025 até às 23h50**

**Data limite de entrega (trab. est.; dirigentes): 10-1-2025 até às 23h50**

### ESTouro

A produtora de jogos, ESTtrenimento, presta homenagem à clássica série de jogos “Bloons Tower Defense”, lançando a sua própria versão – ESTouro. Neste típico jogo de Tower Defense, o jogador tem de colocar torres no meio de um campo de batalha de modo a impedir que os inimigos, que seguem um determinado percurso, cheguem à base do jogador. No caso do ESTouro, os inimigos são balões (bloons) que devem ser estourados pelas torres do jogador. Cada inimigo tem características próprias assim como cada torre. Para colocar uma torre, o jogador tem de a comprar, pelo que é importante gerir o dinheiro disponível. Por cada inimigo eliminado ganha-se dinheiro que pode ser utilizado para comprar novas torres. Por cada inimigo que se deixa passar perde-se, em vidas, o valor desse inimigo.

### Torres

Neste momento são suportadas 6 torres diferentes: macaco, octogonal, canhão, morteiro, balista e ninja. O sistema deve ficar preparado para suportar futuras torres.

Algumas torres são configuráveis. A maneira de as configurar é simples: pressionar na torre e arrastar o rato. Todas elas suportam diversos modos de ataque, embora algumas não os usem diretamente. Para alterar o modo de ataque, basta selecionar a torre e, no painel lateral, escolher o modo de ataque pretendido, usando o botão respetivo. Os modos de ataque afetam o modo como a torre escolhe os bloons a estourar, por forma a otimizar o seu rendimento.

Cada torre dispara um ou mais projéteis. Cada projétil provoca um dado estrago no bloon. Se esse estrago for maior que a resistência restante do bloon este estoura. Os projéteis podem ser de dois tipos: perfurantes ou de explosão. Os perfurantes afetam um bloon de cada vez. Se o estrago for absorvido pelo bloon atingido, o projétil termina o seu caminho. Se o bloon estourar com o projétil e este ainda tiver estrago sobrante, continua o seu caminho até atingir outro bloon ou atingir o alcance máximo. Os projéteis explosivos atingem todos os bloons que estejam no raio de ação da explosão.

De seguida são apresentadas as torres atualmente suportadas.

#### Torre Macaco

Esta torre atira um dardo que provoca um estrago de 2 (pode estourar até 2 bloons simples). A torre tem um raio de ação e apenas começa a atirar dardos quando um bloon entra no seu raio de ação. Faz uso de todos os modos de ataque.

#### Torre Octogonal

Esta torre atira 8 dardos, um em cada direção dos seus lançadores. Cada um dos dardos provoca um estrago de 1. A torre tem um raio de ação e apenas começa a atirar dardos quando um bloon entra no seu raio de ação. Não faz uso dos modos de ataque.

O jogador pode rodar a torre de modo a escolher as direções dos disparos.

#### Torre Canhão

Esta torre atira uma bomba capaz de reduzir em 2 a resistência dos bloons presentes na zona próxima do seu rebentamento. A bomba só rebenta se entrar em contacto com um bloon.

---

A torre tem um raio de ação e apenas começa a atirar bombas quando um bloon entra no seu raio de ação. Faz uso de todos os modos de ataque.

### **Torre Morteiro**

Esta torre atira uma bomba capaz de reduzir em 2 a resistência dos bloons presentes na zona próxima do seu rebentamento. Esta torre não tem raio de ação já que atira as suas bombas sempre para o mesmo local, quer haja bloons ou não. A zona a atingir – área de ataque - pode ser definida pelo jogador e mudada a qualquer momento. Apesar de não ter raio de ação, a torre tem um alcance máximo, isto é, só pode definir a área de ataque dentro de um determinado raio à volta da torre. Não faz uso dos modos de ataque.

### **Torre Balista**

Esta torre atira uma seta enorme com estrago de 10, capaz de estourar vários bloons em linha. Como é pesada, não tem sistema de orientação, pelo que terá de ser o jogador a definir a direção em que ela dispara. Ela dispara sempre que tiver um bloon na sua linha de visão, ignorando assim os vários modos de ataque.

### **Torre Ninja**

Esta torre alterna entre atirar 3 dardos com um estrago de 3 cada, ou uma granada com estrago de 2. A granada só explode se entrar em contacto com um bloon. Esta torre possui vários modos de ataque, configuráveis pelo jogador, que afetam como escolhe os bloons a estourar, por forma a otimizar o rendimento da mesma.

## **Bloons**

Neste momento há suporte a vários bloons diferentes, que podem ser catalogados em 3 categorias: bloons simples, bloons multicamada e bloons fabricantes. O sistema deve ficar preparado para suportar novas categorias de bloons.

### **Bloon Simples**

Bloons que estouram quando sofrem estragos maiores que a sua resistência.

### **Bloon Multicamada**

Bloons que quando recebem maior estrago que a sua resistência estouram, mas quando estouram libertam outros balões que estavam no seu interior.

### **Bloon Fabricante**

Bloons que produzem outros bloons enquanto se movimentam. Tal como os outros bloons, estouram quando recebem maior estrago que a sua resistência.

Pretende-se também dar suporte a estas categorias:

### **Bloon imune a elementos perfurantes**

Este pode ter as características de qualquer das outras categorias, mas tem também a imunidade a projéteis perfurantes. Só pode ser estourado por projéteis explosivos.

### **Bloon imune a elementos explosivos**

Este pode ter as características de qualquer das outras categorias, mas tem também a imunidade aos projéteis explosivos. Só pode ser estourado por projéteis perfurantes.

A tabela seguinte mostra os bloons pretendidos, identificados pela respetiva categoria:

Nome	Categoria	Veloc.	Resist.	Valor	Observações
Vermelho	Simples	3	1	2	
Vermelho rápido	Simples	8	1	2	Não aparece por si só, é apenas libertado por outros.
Azul rápido	Simples	10	2	3	Não aparece por si só, é apenas libertado por outros.
Azul	Multi-camadas	4	2	3	Dá origem a 1 vermelho
Verde	Multi-camadas	4.5	3	4	Dá origem a 1 azul
Amarelo	Multi-camadas	5	4	5	Dá origem a 1 verde e 1 vermelho rápido
Rosa	Multi-camadas	5	4	6	Dá origem a 1 amarelo, 2 vermelhos rápidos e 1 azul rápido
Metal	Multi-camadas Imune Perfurantes	3	5	7	Dá origem a 2 rosas
Barro	Multi-camadas Imune Perfurantes	3.5	4	7	Dá origem a 2 pretos
Preto	Multi-camadas Imune Explosões	4	6	7	Dá origem a 2 rosas
Branco	Multi-camadas Imune Explosões	4	6	7	Dá origem a 2 metal
Zepelim Azul	Fabricante	2	18	12	Cria 50% vermelhos e 50% azuis Ritmo disparo: 30 ciclos
Zepelim Verde	Fabricante	2.3	22	20	Cria 25% vermelhos, 25% azuis e 50% verdes. Ritmo disparo: 30 ciclos
Zepelim Amarelo	Fabricante	2.8	24	25	Cria 33% azuis, 33% verdes e 33% amarelos. Ritmo disparo: 30 ciclos
Zepelim Rosa	Fabricante	3.4	28	30	Cria 17% azuis, 17% verdes, 33% amarelos e 33% rosas. Ritmo disparo: 30 ciclos
Zepelim Metal	Fabricante Imune Perfurantes	2.3	30	45	Cria 50% verdes e 50% amarelos. Ritmo disparo: 30 ciclos
Zepelim Preto	Fabricante Imune Explosões	2.7	30	45	Cria 50% amarelos e 50% rosas. Ritmo disparo: 30 ciclos

## Modos de Ataque

O jogo tem, neste momento, 4 modos de ataque suportados. Quando seleciona uma torre no jogo, o jogador pode escolher qual o modo de ataque para essa torre. Todas as torres possuem um modo de ataque, embora algumas não o usem ativamente. Os modos de ataque permitem escolher um bloon alvo de entre os vários bloons elegíveis. Os modos suportados neste momento são:

### Primeiro

A torre escolhe o bloon que esteja mais à frente no caminho. Útil para evitar que os bloons que estejam mais à frente se escapem.

### Último

A torre escolhe o bloon que esteja mais atrás no caminho. Útil para apontar ao último e, pelo meio, atingir o maior número de bloons possível. Bom especialmente se a torre tiver um ataque poderoso.

### Perto

A torre escolhe o bloon que esteja fisicamente mais perto dela. Útil para evitar que a torre falhe por estar a apontar para alvos distantes, pois como o projétil tem menos espaço a percorrer é mais fácil acertar.

### Juntos

A torre aponta para a zona onde houver mais bloons. Útil quando se quer atingir um grande número de bloons ao mesmo tempo, como quando se usam projéteis explosivos.

## Pistas

O jogo tem, neste momento, 4 pistas. O jogador pode escolher, no arranque, em qual das pistas pretende jogar. Cada pista tem uma série de rounds. Quando termina todos os níveis de uma pista, passa automaticamente para a pista seguinte. As torres não passam de uma pista para a outra, mas o dinheiro angariado sim. Isso significa que se deve tentar usar o menor número de torres nas primeiras pistas para poder ter mais dinheiro nas pistas mais exigentes, já que a dificuldade não é a mesma em cada pista.

A configuração das pistas é dada num ficheiro de configuração.

## Ficheiros de configuração

Existem dois ficheiros de configuração. O objetivo destes ficheiros é garantir que, apesar de se poderem acrescentar novas torres/níveis o código do jogo não é alterado. Os ficheiros de configuração mais importantes são:

### *TowerInfo.txt*

Este ficheiro tem informações sobre cada torre: custo, descrição, etc. Qualquer alteração a estas características deve ser feita aqui. Quando se acrescentam novas torres este ficheiro tem de ser alterado. O jogo lê este ficheiro para saber a ordem e qual a informação que deve disponibilizar sobre as torres nos painéis laterais. O seu leitor é a classe `TorreReader`.

### *Tracks.txt*

Este ficheiro contém a configuração sobre o jogo. Essa informação consiste na quantidade de dinheiro inicial, o número de vidas e a informação sobre as várias pistas, nomeadamente quantos rounds tem cada pista. O seu leitor é o próprio jogo.

## Código fornecido

O código fornecido já suporta a maior parte das características referidas, mas poderia ter sido mais bem desenhado (dizem as más-línguas que foi mal desenhado de propósito!). As soluções arranjadas, apesar de funcionarem, não são muito escalonáveis nem fáceis de manter. Compete ao seu grupo de programadores gerar melhores soluções que as apresentadas. Muitas das soluções, passam pelo uso de patterns (conceito que os autores do anterior sistema não dominavam, ou fizeram por esquecer). Apesar do código estar comentado e assinaladas as questões fundamentais, não se descrever aqui os principais problemas.

### Elevado uso de switch nas torres

O código das torres está demasiado baseado em switch especialmente no que se refere ao modo de ataque. Isso significa que o código é (ainda) mais complicado de se perceber. Todos os switch destas classes terão de ser removidos por soluções que sejam mais expansíveis.

### Muito código repetido nas torres

O código das torres contém muito código muito parecido, com especial enfase no método atacar. Devem optar por uma solução que permita reduzir ao máximo a repetição de código, quer nas torres já existentes quer nas torres futuras.

### Uso do instanceof

Existem vários locais onde se usou o instanceof. O uso do instanceof é revelador de código que não é escalável nem versátil por assumir que apenas alguns tipos de dados são usáveis. Esta construção está assinalada com TODO. Devem eliminar o uso do instanceof em todos estes locais. **NUNCA equacionar a hipótese de adicionar strings/caracteres/inteiros ou outros para codificar o tipo dos elementos.**

### Problema com o Bloon fabricante

O código do bloon fabricante não está a funcionar muito bem, já que ele não está a criar novos balloons, mas a reutilizar os que são usados para a configuração, o que leva a problemas

inesperados, como bloons a andar mais rápido do que deviam, a voltar atrás, entre outras anomalias. Devem corrigir este problema.

## Coisas novas a fazer

O código fornecido está estruturado de forma a minimizar as alterações, caso sejam adicionadas novos tipos de torres ou novos tipos de bloons. Assim, para adicionar novos tipos de torres ou bloons devem ser alteradas as classes TorreCreator e BloonCreator, respetivamente. O modo como resolvem os problemas anteriores deve ter em conta este pormenor. As únicas classes que poderão ser alteradas devido à introdução de novas torres/bloons são: BloonsCreator, TorreCreator, ManipulatorCreator e GameWriter e GameReader. Na criação de novos modos de ataque podem ter de alterar a classe ConfiguradorTorres.

Para testar as soluções anteriores devem ser criadas 2 novas categorias de bloons, uma nova torre e 2 novos modos de ataque.

### Torre Sniper

Esta torre tem a capacidade de atingir um alvo a qualquer distância (tem alcance infinito) e com efeitos imediatos. Dispara um dardo, com estrago 5, que atinge o inimigo imediatamente. Após o impacto com o primeiro inimigo o dardo perde a velocidade inicial e passa a comportar-se como um dardo normal. A maneira mais simples é assumir que o dardo é atirado diretamente de dentro do bloon alvo.

Esta torre dispara sempre na mesma direção e terá de ser o jogador a escolher essa direção. Dispara quando tiver um bloon na sua linha de visão. Faz uso de todos os modos de ataque.

**Nota importante:** Após criar a torre devem alterar o ficheiro *TowerInfo.txt* adicionando a informação sobre a torre. Devem também atualizar a classe TorreCreator.

### Modo de ataque - Longe

A torre ataca o bloon que estiver mais afastado fisicamente dela. Útil para apanhar aqueles bloons que estão quase a sair do alcance.

### Modo de ataque - Forte

A torre ataca o bloon que for o mais valioso, que normalmente são os mais fortes. Não se usa o valor da resistência, pois um bloon vai perdendo a resistência e pode deixar de ser o mais forte. Útil usado em conjunto com várias torres para concentrar os ataques num dos bloons fortes para este ser eliminado o mais rapidamente possível.

### Bloon com armadura temporária

Pode ter as características de qualquer das outras categorias, mas tem também a imunidade a elementos perfurantes, durante alguns impactos. Após esses impactos perde a armadura.

### Bloon com escudo temporário

Pode ter as características de qualquer das outras categorias, mas tem também a imunidade aos elementos explosivos, durante algumas explosões. Após essas explosões perde o escudo.

Por forma a testar estas características devem fazer as alterações nos bloons indicados na tabela seguinte:

Nome bloon	Categoria	Veloc.	Resist,	Valor	Observações
Zepelim Metal	Fabricante Imune Perfurantes	2.3	30	45	<b>Os balões criados têm a armadura temporária durante 8 impactos.</b> Cria 50% verdes e 50% amarelos. Ritmo disparo: 30 ciclos
Zepelim Preto	Fabricante Imune Explosões	2.7	30	45	<b>Os balões criados têm o escudo temporário durante 12 impactos.</b> Cria 50% amarelos e 50% rosas. Ritmo disparo: 30 ciclos

## Objetivos

O objetivo deste trabalho é alterar o sistema tal como foi descrito, dando-se especial atenção à correta implementação da arquitetura do sistema, ao uso correto das patterns estudadas aplicáveis e às características referidas no desenvolvimento de software. O tratamento de erros poderá levar à criação de exceções específicas. Devem criar as que forem necessárias, e apenas essas.

Sugestão dos passos a seguir	Critérios de avaliação	Conceitos a utilizar
Analisar os problemas apresentados dando grande atenção ao código fornecido; Identificar quais as patterns que ajudam neste trabalho Implementar as várias patterns uma a uma. Criar a nova torre Criar os novos modos de ataque Criar os novos tipos de bloons	Interligação entre os objetos Implementação das patterns Funcionamento do sistema Relatório final Defesa do trabalho	Herança Exceções (se aplicável) Análise de problemas Relações entre classes Patterns

### Nota:

A otimização de algumas das operações será beneficiada. Por outro lado, soluções pouco eficazes (desperdício de memória exagerado, código redundante, etc) serão penalizadas.

O código deve seguir as regras de codificação estabelecidas na disciplina.

### O que se deve entregar:

- As regras para entrega serão anunciadas mais tarde.

### Dúvidas:

Eventuais dúvidas podem ser esclarecidas junto dos docentes da disciplina. Podem também usar a plataforma de e-learning, nomeadamente o fórum.

**Os trabalhos ficarão sujeitos a uma defesa em data a marcar após a entrega dos trabalhos.**

**Trabalhos idênticos serão anulados.**