

# Laboratórios de Informática II

## *Geração de Tabuleiros com Solução Única*

Nelson Miguel de O. Estevão A76434

Pedro Alexandre Gonçalves Ribeiro A85493

Rui Filipe Moreira Mendes A83712

Junho 2018

## Introdução

No âmbito da Unidade Curricular *Laboratórios de Informática II* foi-nos proposto a criação de um programa capaz de gerar tabuleiros para o jogo - **GandaGalo** - desenvolvido durante esta mesma Unidade Curricular.

Os tabuleiros deste jogo podem conter três tipos de peças, tal como se pode verificar na figura seguinte:

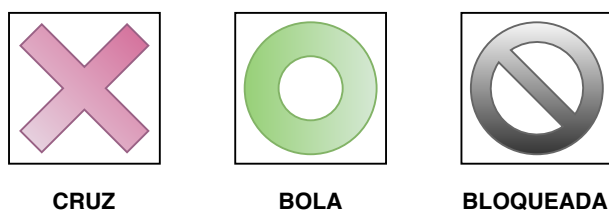


Figura 1: Tipos de Peças

Os tabuleiros são matrizes que podem ou não ser quadradas. Estes estão compreendidos entre 1 a 20 quadriculas (tanto linhas como colunas). O objetivo é completar o tabuleiro, ou seja, preencher todas as quadriculas com uma *cruz* ou com uma *bola*. No final, não podem existir mais de duas bolas nem duas cruzes seguidas em qualquer uma das direções, horizontal, vertical e diagonal.

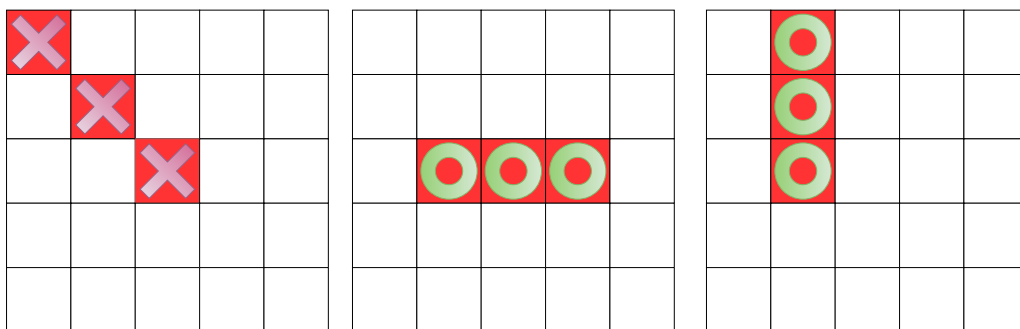


Figura 2: Exemplos de casos inválidos

Não existe qualquer restrição para as peças bloqueadas, ou seja, não existe qualquer limitação de quantas podem estar seguidas. As bloqueadas interrompem as sequências de cruzes ou bolas na sua intersecção. Contudo, não é permitido ao jogador colocar uma peça bloqueada. Estas existem num tabuleiro ou não no início e nenhuma outra pode ser acrescentada durante.

Estas devem ser colocadas estrategicamente em locais que nenhuma outra seria possível.

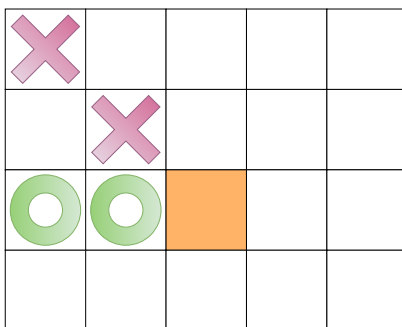


Figura 3: Tabuleiro impossível de resolver

Qualquer tabuleiro que tivesse uma configuração inicial do tipo representado na Figura 3 seria de resolução impossível, uma vez que, quer se coloque uma bola na casa laranja quer se coloque cruz passaria a estar num dos casos representados na Figura 2. Um tabuleiro não é impossível de resolver se apresentar uma situação de resolução impossível logo à partida mas sim, se a sua configuração leve a uma serie de jogadas forçadas que resultam num caso impossível. No caso em concreto, passaríamos a estar num caso válido tivesse uma peça bloqueada nesse lugar, tal como está representado na Figura 4.

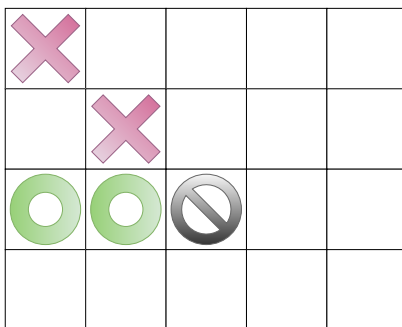


Figura 4: Correção possível para tabuleiro impossível

Contudo, um tabuleiro só é verdadeiramente válido se, para além de ser de possível resolução, tiver uma e uma só possível solução. Este relatório descreve como conseguimos criar um programa capaz de gerar tabuleiros com uma única solução recebendo apenas a dimensão desejada e a dificuldade esperada.

## Implementação

Uma das primeiras características que o nosso programa precisava de ter era uma certa aleatoriedade. Não seria de todo interessante se dados os mesmos argumentos, obtivéssemos o mesmo tabuleiro.

```
int x = rand() % 10;  
int y = rand() % 10;
```

Este programa, apelidado de **gerar**, recebe três argumentos:

- dificuldade (1 para fácil, 2 para difícil);
- número de linhas (até 20);
- número de colunas (até 20).

## Conclusão

Fazendo uma pequena análise relativa às competências adquiridas durante a realização deste trabalho conseguimos concluir que o mesmo foi bastante importante no que diz respeito à aquisição de conhecimentos sobre algoritmos, tendo sido a nossa primeira introdução séria a essa área.

Achamos também que alcançamos os objetivos desta fase do trabalho, ou seja, conseguimos gerar com sucesso tabuleiros resolúveis de dificuldades distintas, conforme a vontade do utilizador.

De uma maneira geral é nossa convicção que atingimos os objetivos propostos aquando da apresentação deste trabalho.