Instituto Superior Técnico 2016/2017

**Programação Orientada por Objetos**

**Relatório do projeto – Vídeo Poker**

Realizado por:

Manuel Monteiro nº 79138

João Amado nº 79185

João Heleno nº 78921

Miguel Paulino nº 79168

**Introdução**

É pretendido neste trabalho final da unidade curricular de Programação Orientada por Objetos implementar na linguagem de programação java o jogo *Vídeo Poker*, similar ao jogado nas *slot machines* dos casinos.

Resumidamente o *Vídeo Poker* consiste num jogo de cartas, jogado com um baralho normal de 52 cartas. Após uma aposta inicial são dadas ao jogador cinco cartas, das quais ele pode optar por descartar uma ou mais, caso este opte por o descarte as respetivas cartas serão substituídas por outras pertencentes ao restante baralho. Por ultimo é avaliada a mão do jogador e este é pago consoante o seu *ranking*. No desenvolvimento deste trabalho teve-se em conta todos os detalhes do jogo de modo a torna-lo mais realístico possível.

**Detalhes de Implementação / Estrutura**

Após a leitura do projeto foi feita uma avaliação de modo a identificar quais as classes fundamentais ao seu desenvolvimento. Começou-se então por definir as classes mais óbvias, uma carta, de seguida o baralho que representa um conjunto de cartas e por sua vez a mão, que constitui um conjunto de cartas retiradas do baralho.

De seguida optou-se então por criar outras duas classes importantes, o jogador e o dealer. Uma é responsável por armazenar e atualizar vários dados acerca do jogador, como a sua mão, o seu dinheiro, as suas cartas e estatísticas de jogo. Enquanto a outra é responsável pela distribuição de cartas, pelos pagamentos e o embaralhamento do *deck*.

Tendo então os fundamentos base implementados, resta-nos então desenvolver o mecanismo de jogo tendo em conta as suas regras. São propostos então três modos diferentes de jogo: o modo interativo, o de *debug* e o de simulação. O primeiro modo permite que o jogo seja jogado através do terminal enquanto que no segundo o jogo é executado através de um ficheiro auxiliar. Por ultimo no terceiro modo o jogo é jogado automaticamente de acordo com a estratégia perfeita para fácil visualização dos ganhos médios.

Dada a similaridade entre os diferentes modos de jogo decidiu-se então tirar partido de uma classe abstrata, *AbstractGame*, dos quais os modos de jogo *interative* e *degub* são herdeiros. Abstrata neste caso pois independentemente do modo de jogo os métodos implementados são os mesmos, mas com ligeiras mudanças. As etapas comuns aos diferentes modos são então a fase de *bet*, *deal*, *hold* e *evaluation*. Foi também criada uma interface de jogo, Game, de modo a ser possível utilizarmos os diferentes métodos de uma maneira genérica independentemente do modo de jogo.

Como ponto de partida, começou-se por desenvolver o modo interativo em simultâneo com as classes relativas às pontuações das mãos. Graças ao idêntico funcionamento com o modo interativo, tendo este desenvolvido, o modo *debug* tornou-se bastante simples de implementar, visto que a diferença está no modo como o programa recebe os comandos de jogo, por via terminal ou por via ficheiro .txt. Quanto ao modo de simulação, a sua complexidade de implementação foi maior, pois é necessário que o programa escolha quais cartas manter na mão de forma a obter uma maior pontuação. Este foi então desenvolvido em paralelo com o comando *advice*, que ajuda o jogador a escolher quais cartas descartar de modo a otimizar o seu ganho.

**Interface Gráfica – Bonus**

**Resultados / Conclusões**