

# Sistemas Operativos

Trabalho Prático - “CHAMPION”

João Pedro Verdete Santos	2017011382
Rafael de Jesus Saraiva	2017010339

## Pormenores de Implementação

### Árbitro.c

Em relação, à obtenção de argumentos, é usado o *getopt()* para limitar o uso das argumentos que são pedidos, rejeitando todos os outros. Nesta fase, o árbitro só apresenta as variáveis que obtém para *debugging*.

### Constantes.h (árbitro)

Foram e serão definidas (ao longo das metas) variáveis de erro com códigos associados a cada uma, para simplificação, esses códigos estão definidos geralmente pelo programa. Por exemplo, caso não seja detectada uma variável de ambiente, vai ser escrito no código *return ERRO\_FALTA\_ENV*, que resulta num *return 11*, sendo mais fácil a leitura das saídas do programa/funções.

## Principais Opções Tomadas

### Makefile

Decidiu-se “automatizar” o makefile para os programas deste trabalho, com esta “automatização” referimo-nos a excluir por completo a necessidade de alterar este ficheiro (makefile) caso se acrescentem novos ficheiros fonte/headers aos programas.

Foram usadas variáveis para fácil mudança de pastas de compilação/fontes, bem como para linhas de *debugging* (uso de cores).

Para a detecção automática dos jogos, usou-se uma regra “dinâmica”, quer dizer que aceita um formato para a accionar, por exemplo, *jogo-001* compila o ficheiro *jogo-001.c*, *jogo-002* compila o ficheiro *jogo-002*, sendo o formato *jogo-[variável]*.

Nas regras de compilação do cliente e árbitro, o makefile faz a compilação dos ficheiros fonte apropriados à regra.