Relatório PCD – Agar IO

1. **Colocação inicial dos jogadores:** No momento de colocação inicial dos jogadores, detetámos o problema de se um jogador for colocado na posição de outro, então este nunca iria ser colocado, ficaria à espera eternamente. Desta forma para resolver este problema utilizámos variáveis condicionais na função que coloca o jogador e na função de movimentar o jogador. Na primeira, bloqueamos no caso de já existir um jogador nessa posição e na segunda quando o jogador sai dessa posição dá sinal que está já se encontra livre para o próximo ser colocado.
2. **Resolução da imobilização no movimento:** Relativamente a este ponto tivemos um problema de deadlocks. Isto acontecia no momento em que um player tentava aceder a um outro player que estava já bloqueado anteriormente pelo wait de 2 segundos, isto criava uma situação em que todos os intervenientes desta situação ficassem presos à espera eternamente pelo interrupt mas sem sucesso. A forma como solucionámos foi repensar e reposicionar os locks no método movePlayer de modo a evitar este problema. Após diversos testes acreditamos que o problema ficou resolvido. Foi um passo importante na aprendizagem de como os locks e unlocks funcionam.
3. **Fim do jogo**: Na altura em que terminamos o jogo, quando pedimos às threads para serem interrompidas foi necessário adicionar um booleano de forma a impedir que estas se mexessem de modo a serem eliminadas de forma natural dado que estas como vão sendo interrompidas à medida que existem threads disponíveis para esse trabalho, dava uma sensação estranha ao jogo.
4. **Implementação dos jogadores humanos com aplicações remotas:** Relativamente a este ponto, tentámos reduzir ao máximo o tamanho da aplicação cliente, permitindo ainda ao servidor avisar qual o tamanho do mapa de jogo. Verificámos que quando o jogo terminava, ainda era possível do lado do cliente tentar enviar dados. Para resolver este problema do lado do servidor enviamos a alertar que o jogo foi terminado de modo ao cliente caso pretenda fazer mais alguma coisa ficar a saber explicitamente que o jogo terminou. (Através de um booleano gameOver). Após envio da parte do servidor, fechamos a ligação. Do lado cliente, após receber o booleano valida se de facto o jogo terminou. (Pode ser utilizado para outras coisas, foi criado apenas para o cliente saber que o jogo terminou e que não foi uma falha na rede).

Data: 10/12/2022

Nomes:

Alexandre Monteiro 90284

Nuno Medrôa – 88036

Turma:

IGE-PL-C1