2025

Uma imagem com Tipo de letra, logótipo, Gráficos, texto

Os conteúdos gerados por IA poderão estar incorretos.

Relatório Sistemas Operativos 2

Licenciatura engenharia informática

Martim Oliveira Nº2022132041 Ricardo DUarte Nº2022137878

Índice

[Introdução 2](#_Toc199450155)

[Programas 3](#_Toc199450156)

[Estruturas de dados 3](#_Toc199450157)

[Arbitro 4](#_Toc199450158)

[Jogoui 6](#_Toc199450159)

[Bot 7](#_Toc199450160)

[Tabela de Conteúdos 8](#_Toc199450161)

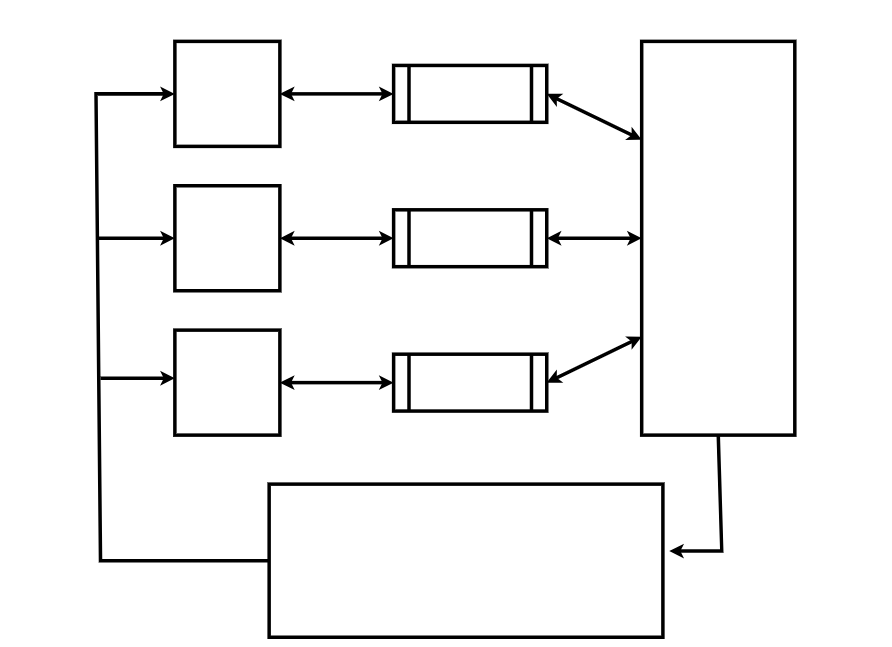
[Conclusão 9](#_Toc199450162)

# Introdução

O relatório descreve o desenvolvimento de um trabalho prático no âmbito da unidade curricular de Sistemas Operativos 2, que consiste na implementação de um jogo de palavras. Embora o foco principal não seja o jogo em si, mas sim na correta aplicação dos conceitos e mecanismos estudados na disciplina, a construção deste projeto permitiu explorar de forma prática a comunicação entre processos, gestão de concorrência, sincronização e controlo de execução num ambiente multi-processo.

A lógica do jogo baseia-se na identificação de palavras válidas a partir de letras apresentadas aleatoriamente no ecrã, a uma cadência constante. Os jogadores interagem com o sistema através de comandos na consola, podendo formar palavras, consultar a pontuação, visualizar outros jogadores ou abandonar o jogo. O jogo suporta até 20 jogadores simultâneos e é gerido de forma automática, sendo necessário a presença de dois ou mais jogadores e terminando quando resta apenas um.

Este relatório descreve as principais funcionalidades implementadas, as decisões técnicas tomadas e os desafios enfrentados durante o desenvolvimento do projeto, destacando a aplicação prática dos conceitos teóricos abordados ao longo da disciplina.



bot

joguiUma imagem com diagrama, esboço, Esquema, Desenho técnico

Os conteúdos gerados por IA poderão estar incorretos.

jogoui

arbitro

Memória partilhada

pipe

pipe

pipe

# Programas

Para a elaboração do jogo, efetuamos o desenvolvimento de 3 programas, nomeadamente “arbitro”, “bot” e “jogoui”, cada um com características distintas.

O programa “arbitro” é responsável pela gestão do jogo, controlando a lógica principal. O “jogoui” corresponde ao cliente utilizado pelos jogadores humanos, servindo de interface para interação com o jogo. Por fim, o “bot” representa um jogador automático que interage com o jogo, imitando o comportamento de um utilizador

## A computer screen shot of a program code AI-generated content may be incorrect.Estruturas de dados

Aqui estão as estruturas usadas no árbitro e no jogoui, respectivamente, são bastante similares, mas as estruturas de controle têm diferenças cruciais, as estruturas de dados do bot são exatamente iguais às do jogoui com exceção da estrutura extra que tem para se organizar.

A black background with green text

AI-generated content may be incorrect.

## Arbitro

O “arbitro”, como o nome indica é quem controla o jogo, sendo assim, o programa mais importante.

Este é único, sendo assim, este garante que, caso alguém tente adicionar outro árbitro seja impossível. Para isso, criamos um mutex, este não permite a criação de outro programa arbitro, caso já seja existente na máquina

Uma imagem com texto, captura de ecrã, Tipo de letra

Os conteúdos gerados por IA poderão estar incorretos.

Após criar este mutex pode então prosseguir para o funcionamento do jogo, começa por chamar a função “initMemAndSync” que cria e inicializa a memória partilhada, chama uma função que vai ao registry buscar os valores para o número máximo de letras e o ritmo e também inicializa alguns campos da estrutura de controle que vai ser usada ao longo do programa, é também criado um evento que é usado para alertar os clientes que tem letras novas na memória partilhada.

A computer screen shot of a program code

AI-generated content may be incorrect.

De volta no main, é criada a thread que vai tratar de todos os comandos inseridos pelo árbitro, comandos estes como listar jogadores, excluir jogadores e criar bots com um nome inserido pelo árbitro.

A computer screen with text

AI-generated content may be incorrect.

Finalmente entramos no ciclo principal do programa em que são criados tantos named pipes e threads para tratar da comunicação quantos clientes se juntarem ao jogo, são primeiro verificados os nomes inseridos pelos clientes para certificar que não há jogadores duplicados, o jogo efetivamente só começa quando se juntarem pelo menos dois jogadores.

Quando se juntarem jogadores suficientes é lançada a thread que vai escrever na memória partilhada e ativar o evento para os clientes.

A computer screen with text

AI-generated content may be incorrect.

O árbitro permanece neste ciclo enquanto não houver sinal de shutdown, quando este for acionado, sai do ciclo, espera que a thread do teclado termine e fecha o programa.

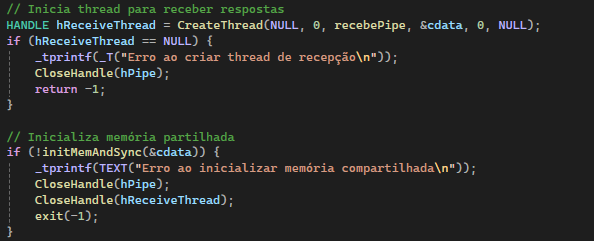
## Jogoui

O jogoui é bastante mais simples, recebe o seu nome por argumento e imediatamente tenta ter conexão com o pipe para tantar fazer login no jogo, se for bem-sucedido prossegue com o jogo, se não fecha imediatamente.

A computer screen with many colorful text

AI-generated content may be incorrect.

Depois de um login bem-sucedido é lançada a thread recebePipe que vai receber respostas a comandos inseridos por si, vindos do árbitro, tem, tal como o árbitro uma função initMemAndSync que inicializa a memoria partilhada e algumas partes da estrutura de controle geral.

Lança também logo de seguida a thread comunica que vai ler, em loop, o teclado, verificar se foi inserido um comando ou uma tentativa de adivinhar palavra, e envia o que tiver sido inserido (comando ou palavra) para o árbitro, este lê e responde apropriadamente.

Finalmente é aberto o evento para leitura da memória partilhada e é lançada a thread que a vai ler e apresentar no ecrã sempre que houver alterações.

A screen shot of a computer program

AI-generated content may be incorrect.

## Bot

Como dito anteriormente o programa bot é semelhante ao programa jogoui, no entanto este não qualquer interação com o utilizador, sendo “lançado” e funcionando sozinho. Este quando é executado necessita de dois argumentos adicionais, o seu “nome” e a velocidade que deve enviar as palavras.

A comunicação com o programa árbitro é igual ao jogoui, ou seja, sabe as letras através da memória partilhada e envia palavras através de named pipes.

A diferença deste programa para o jogoui, é que este efetua o lançamento das palavras automaticamente, para isso, este sabe o dicionário de palavras possíveis e envia uma dessas palavras.

Uma imagem com texto, captura de ecrã, software

Os conteúdos gerados por IA poderão estar incorretos.

# Tabela de Conteúdos

|  |  |  |
| --- | --- | --- |
| ID | Descrição funcionalidade / requisito | Estado |
| 1 | As letras são apresentadas de forma aleatória, a uma cadência fixa | implementado |
| 2 | Trocar a letra mais antiga | implementado |
| 3 | Sistema de pontos | Implementado |
| 4 | Remover letras do array | Implementado |
| 5 | Limite users | Implementado |
| 6 | Número mínimo para começar | Implementado |
| 7 | Verificação de users | Implementado |
| 8 | Informação entre processos/programas | Implementado |
| 9 | Diferença entre comando e palavras | Implementado |
| 10 | Informação de jogo/participantes | Implementado |
| 11 | Manipulação de utilizadores | Implementado |
| 12 | Velocidade de jogo | Implementado |
| 13 | painel | Não Implementado |
| 14 | Uso registry | Implementado |
| 15 | Named pipdes | Implementado |
| 16 | Threads | Implementado |
| 17 | Mutex | Implementado |
| 18 | Memória Partilhada | Implementado |

O painel não foi implementado, devido a sua complexidade e tempo necessário, para efetuar testes, corrigir bugs, entre outros problemas adjacentes

# Conclusão

A realização deste projeto permitiu aplicar, diversos conceitos fundamentais da unidade curricular de Sistemas Operativos 2, como memória partilhada, sincronização com mutexes e eventos, comunicação interprocessos através de named pipes e utilização de threads.

Apesar de o foco não ter sido na complexidade do jogo em si, foi possível desenvolver uma aplicação funcional e estável, com suporte para múltiplos jogadores e interações em tempo real. A separação em três programas distintos (“arbitro”, “jogoui” e “bot”) evidenciou a importância da coordenação entre processos e a gestão rigorosa da partilha de recursos.

Em suma, este trabalho constituiu uma oportunidade valiosa para consolidar conhecimentos sobre os mecanismos de baixo nível do sistema operativo Windows.