

### Nome dos integrantes:

Tiago de Jesus Chehab – 2211194

Lucas Santana de Souza – 2211131

Gabriella Stefany Silva Pereira - 2211288

# Relatório - Trabalho Prático Hovercooked

## Introdução

Hovercooked é uma simulação de um jogo de restaurante onde os jogadores devem preparar e servir pratos em um ambiente de cozinha. Este documento detalha a implementação do jogo utilizando a concorrência dentro da linguagem de programação C. O jogo simula uma cozinha onde pedidos são gerenciados e preparados por cozinheiros. A implementação utiliza threads, seções críticas e sincronização para garantir o funcionamento correto do jogo.

## Mecânica do Jogo

O jogo se desenrola em um loop, onde novos pedidos são adicionados de tempos em tempos a um mural de pedidos que devem ser concluídos antes do tempo limite acabar. Os jogadores controlam os cozinheiros, atribuindo-lhes pedidos específicos. Cada cozinheiro, representado por uma thread individual, processa um pedido de cada vez, seguindo as etapas de preparação e cozimento para depois concluir o mesmo.

A cozinha possui alguns recursos, como bancadas e fogões, na qual os cozinheiros precisam compartilhar e não podem ser usados simultaneamente por dois ou mais cozinheiros. O jogo possui diferentes níveis de dificuldade, na qual irá desafiar o jogador de forma progressiva.

## Threads e Concorrência

Thread de mural de pedidos (*mural\_pedidos*): Essa thread é responsável por gerar novos pedidos e adicioná-los ao mural de pedidos num intervalo de 1 a 5 segundos. A geração de pedidos é aleatória.

Thread de gerente (*gerente*): Essa thread lê comandos do usuário para atribuir pratos aos cozinheiros.

Thread de cozinheiro (*cozinheiro*): Cada cozinheiro é uma thread que espera um pedido, prepara os ingredientes, prepara o prato e libera os recursos.

## Estrutura do Projeto

O projeto está dividido nos seguintes arquivos:

- main.c: Função principal e inicializações.
- cozinheiro.c: Implementação das funções relacionadas aos cozinheiros.
- gerente.c: Implementação das funções relacionadas ao gerente.
- mural\_pedidos.c: Implementação das funções relacionadas ao mural de pedidos.
- tela.c: Implementação das funções relacionadas à interface ncurses.
- hovercooked.h: Definições de estruturas e declarações de funções

Compilação e Execução Requisitos GCC Biblioteca ncurses Compilação Para compilar o projeto, utilize o comando make: sh Copiar código make Execução Para executar o projeto, utilize o comando: sh Copiar código ./hovercooked

## Compilação e Execução Requisitos

### 1. Pré-requisitos:

- Sistema operacional Ubuntu.
- Compilador GCC.
- Biblioteca ncurses.

### 2. Instalar as Bibliotecas Necessárias:

- “sudo apt-update”
- “sudo apt-get install libncurses5-dev libncursesw5-dev”

### 3. Compilar código:

◦ Para compilar o projeto, utilize o comando make: sh Copiar código make Execução Para executar o projeto, utilize o comando: sh Copiar código ./hovercooked

### 4. Executar o Jogo:

- ./hovercooked

## Dificuldade do Jogo

Ao iniciar o jogo você terá que escolher a dificuldade do mesmo, podendo selecionar entre Fácil, Médio e Difícil com a complexidade do aumentando de forma gradativa.

- Fácil: Você terá apenas um pedido e sem tempo limite.
- Médio: Você terá todas as bancadas e todos os cozinheiros a sua disposição.
- Difícil: Você terá uma bancada a menos para concluir seus pedidos.

Ao escolher o nível Médio ou Difícil, insira o tempo de jogo em segundos.

## Controles do Jogo

Na interface do jogo é necessário que se use os seguintes comandos para gerenciar os cozinheiros:

- Cozinheiro 1: 11 (Pizza), 12 (Hambúrguer), 13 (Suco)
- Cozinheiro 2: 21 (Pizza), 22 (Hambúrguer), 23 (Suco)
- Cozinheiro 3: 31 (Pizza), 32 (Hambúrguer), 33 (Suco)
- Cozinheiro 4: 41 (Pizza), 42 (Hambúrguer), 43 (Suco)

É importante destacar que cada cozinheiro irá para uma bancada disponível de forma automática. Caso não tenha uma bancada disponível, o cozinheiro vai a alguma bancada após ela ficar livre.

Cada prato exige um certo tempo de preparo para ser finalizado, no qual irá consumir tempo na bancada:

- Pizza: 5 segundos.
- Hambúrguer: 3 segundos.
- Suco: 2 segundos.

Os pedidos chegam ao gerente (jogador), através do mural de pedidos que fica localizado à esquerda da tela do jogo, os pedidos irão chegar de forma aleatória. Após um cozinheiro concluir um pedido, irá aparecer no local onde ficava a demanda uma mensagem de concluído.

É necessário concluir todos os pedidos que irão chegar, antes do tempo determinado pelo jogador acabar. Caso não tenha sucesso, o usuário será notificado que a partida foi perdida e que o objetivo não foi concluído.

## Conclusão

Este projeto demonstra a aplicação prática de conceitos de programação concorrente para simular um ambiente complexo como uma cozinha de restaurante.

A utilização de threads, mutexes, variáveis de condição e semáforos permite a coordenação eficiente de múltiplos cozinheiros e o compartilhamento de recursos limitados. A interface baseada em texto, criada com a biblioteca ncurses, oferece uma representação visual clara do estado do jogo, tornando a experiência mais interativa e compreensível.

Caso queira acessar o repositório criado pelos alunos que desenvolveram o projeto é só acessar o link: <https://github.com/Gabriella-gssp/OvercookedThreads>.