

**UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL
FACOM-UFMS**

Nome: Vinícius Fontes de Andrade



Algoritmos e Programação II

**Trabalho Prático
Sudoku**

Sumário

Introdução

Decisões de Implementação

Funcionalidades não Implementadas

Problemas Enfrentados na Implementação

Descrição do Ambiente de Desenvolvimento

Referências Bibliográficas

Introdução

O trabalho desenvolvido tem como objetivo o desenvolvimento do jogo “Sudoku”. Utilizou-se a linguagem C++ para implementar as funcionalidades exigidas, com o código sendo editado e executado no Visual Studio Code (VSCode). Durante o desenvolvimento, foi dada atenção à organização do código, eficiência e clareza na implementação das funcionalidades, visando principalmente a otimização das funções, levando em conta o uso de arquivos tanto binários, quanto de texto.

Decisões de implementação

- **Estrutura de Dados:** Foram escolhidos vetores, laços, abertura e fechamento de arquivos, posicionamento de ponteiros, métodos de escrita e leitura nos mesmos, de forma simplificada, manipulação de strings, operações matemáticas, entrada/saída de dados, testes constantes a cada implementação feita.
- **Ferramentas utilizadas:** Utilizou-se o GitHub como maneira de administrar melhor a organização do ambiente do trabalho, foram revisados os slides de arquivos binários e arquivos de texto, funções como fseek e fwrite também passados em aula, leitura de um PDF do livro (DEITEL, P.; DEITEL, H. C: Como Programar. 6ª ed. São Paulo: Pearson, 2011;), o compilador do VSCode/Terminal também foi um ótimo aliado para a verificação de testes e erros do código.

Funcionalidades Não Implementadas

Não houve nenhuma funcionalidade que não teve sua devida implementação, todas as funções foram enfrentadas de forma que o trabalho corresse de forma objetiva.

Problemas Enfrentados na Implementação

Os principais problemas enfrentados foram as funções:

1- FILE* crie_arquivo_binario(char quadro[9][9]):

No qual, a parte de gerar um arquivo randômico foi simples, porém a parte de posicionar os ponteiros do arquivo para o começo, escrever o número de jogadas, posicionar o ponteiro para 4 bytes posteriores e depois escrever no quadro, foi bem dificultoso, pois eu estava bem perdido com o que realmente deveria ser feito, porém com pesquisas e revisão dos slides, consegui implementar a função corretamente.

2- void salve_jogada_bin(FILE *fb, char quadro[9][9]):

Função no qual, tive bastante dúvida ao implementar, pois eu tinha entendido sua implementação de uma forma diferente e acabou que eu tinha ido para um caminho totalmente errado, usando um For para escrever a cada vez no arquivo de forma errônea, porém após uma análise e leitura maior do PDF, pude entender melhor a proposta e executar as formas de leitura, posicionamento e escritura do arquivo perfeitamente.

3- FILE *carregue_continue_jogo(char quadro[9][9], char *nome_arquivo) :

A primeira parte desta função foi executada de forma bem simples, porém o Else desta função foi algo que também tive dificuldades na implementação, pois não tinha certeza em que local eu usaria o printf para printar o número de jogadas realmente, logo após, no final consegui entender bem a parte de leitura e posicionamento do arquivo.

Descrição do ambiente de desenvolvimento

- **Compilador:** Foi utilizado o compilador **GNU Compiler Collection (GCC)** versão g++ 11.x.
- **IDE/Editor:** O código foi desenvolvido no Visual Studio Code (VSCode), para facilitar o desenvolvimento em C++, foi utilizando bibliotecas como:
 - #include <stdlib.h>
 - #include <time.h>
 - #include <string.h>
 - #include <stdio.h>
- **Sistema Operacional:** O trabalho foi realizado em Windows (casa), Linux Debian(Laboratório).

Referências Bibliográficas

DEITEL, P.; DEITEL, H. C: Como Programar. 6ª ed. São Paulo: Pearson, 2011;
