

TP - AEDS1

Aluno: Bruno Maletta Monteiro

Matrícula: 2017015150

Data: 21/06/2017

1. Introdução

1.1. Tema

O tema do projeto é o desenvolvimento de um jogo já existente, chamado 2048, porém extendido para base N . O jogo consiste em um tabuleiro, de tamanho K por K , e números que aparecem nele. O jogador pode executar movimentos, os quais podem ser cima, baixo, esquerda ou direita. O movimento faz com que todos os números sejam “puxados” na direção escolhida.

A base N , além de delimitar quais números surgem no tabuleiro, satisfaz a seguinte propriedade: ao se fazer um movimento, N números iguais alinhados na direção escolhida se “juntam”, formando um novo número que é igual à soma deles. Assim, é possível criar números maiores. Por *default*, $K=2N$.

O objetivo do jogo é criar números maiores, gerando uma pontuação maior. O jogador perde o jogo quando o tabuleiro inteiro está completo e não há mais movimentos possíveis.

A pontuação é calculada pela soma de todos os números presentes no tabuleiro dividido pelo tamanho K do mesmo.

O projeto tem funcionalidade inteiramente no terminal do sistema operacional GNU/Linux, não havendo nenhum tipo de interface gráfica. As funcionalidades do jogo se dão pela interação do usuário com o terminal, ao selecionar opções por meio de entradas do teclado.

1.2. Objetivo do Projeto

Desenvolvimento do jogo, como descrito. Criação de menu de opções, com scoreboard, opção de ajuda, que explica como o jogo funciona, opção de continuar jogo salvo. Criação de opção de desfazer último movimento durante o jogo.

1.3. Justificativa da Escolha do Tema

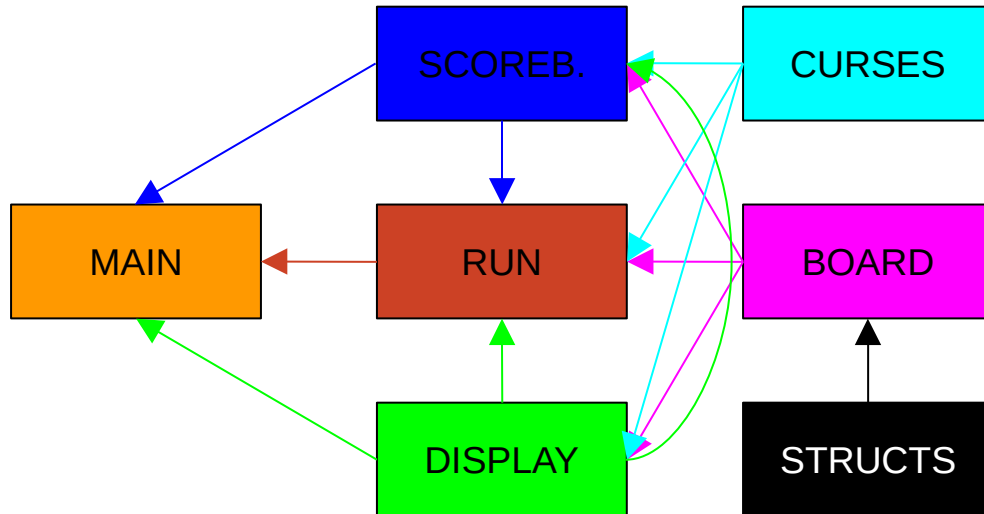
Escolhi o tema tratado porque é um jogo interessante, divertido, de simples implementação e que aborda vários conceitos estudados nas aulas de Algoritmos e Estruturas de Dados I, tais como: arquivo, modularização, structs.

1.4. Organização do Trabalho

O documento está organizado da seguinte forma: há um executável, 2048.exe, que é o arquivo compilado. Há também um pasta src, na qual se encontram os códigos-fonte. Além disso, existem arquivos ocultos .txt, que são os arquivos usados pelo programa para salvar informações, como o scoreboard.

2. Módulos

Diagrama dos módulos do projeto (seta indica inclusão)



2.1. Módulo structs

Encarregado de conter todos os structs usados no projeto. No caso, há apenas um struct, o Board.

2.1.1. Struct Board

Contém:

- **base**: valor inteiro que armazena a base N do tabuleiro;
- **size**: valor inteiro que armazena o tamanho K do tabuleiro;
- **free**: valor inteiro que armazena o número de espaços em branco presentes no tabuleiro;
- **score**: valor de ponto flutuante que armazena a pontuação referente àquele tabuleiro;
- **type**: valor inteiro que indica o tipo do jogo: 0 para médio, 1 para difícil, 2 para profissional e 3 para personalizado;
- **tiles**: matriz de inteiros que representa os valores do tabuleiro.

2.2. Módulo board

Lida com as ações relacionadas ao tabuleiro. Contém as seguintes funções:

- **size**: recebe o valor inteiro da base e retorna o valor do tamanho inteiro correspondente do tabuleiro, que, por default, é duas vezes o valor da base;
- **new_board**: recebe os valores inteiros do tipo do jogo, base e tamanho do tabuleiro, e retorna um tabuleiro tabuleiro vazio correspondente;
- **mirror**: recebe um tabuleiro e retorna o mesmo tabuleiro espelhado, ou seja, com as linhas invertidas;
- **transp**: recebe um tabuleiro e retorna o mesmo tabuleiro transposto.

2.3. Módulo curses

Biblioteca pre-existente, adicionada para o uso da função `getch()`, que lê um caractere do usuário sem que ele tenha que pressionar o Enter.

2.4. Módulo display

Faz as funções referentes a exibir as saídas para o usuário pelo terminal, como exibir o tabuleiro, limpar a tela, etc. São as funções nele contidas:

- **clear_screen**: limpa a tela;
- **draw_file**: recebe uma string com o nome de um arquivo de texto e exibe o texto presente no arquivo;
- **draw_how_to_play**: chama a função `draw_file` para o arquivo “.how_to_play.txt”, que contém o texto que explica o jogo ao usuário;
- **draw_run_menu**: chama a função `draw_file` para o arquivo “.run_menu.txt”, que contém o menu que é exibido durante o jogo;
- **draw_menu**: chama a função `draw_file` para o arquivo “.menu.txt”, que contém o menu inicial;
- **draw_diff**: chama a função `draw_file` para o arquivo “.dif.txt”, que contém o menu de dificuldades;
- **draw_number**: recebe um valor inteiro e a base do jogo (inteira), e exibe o valor, colorindo-o com base na potência da base que ele representa (logaritmo);
- **draw_line**: exibe uma linha de tamanho recebido como argumento;
- **draw_tiles**: recebe um vetor de inteiros, o tamanho do vetor e a base do jogo (inteiros) e exibe os valores do vetor separados por uma barra “|”, mantendo o espaço entre cada barra constante e igual ao valor definido `NUMBER_SPACE`, que, por default, é igual a 5. Chama a função `draw_number` para exibir os números;

- ***draw_board***: recebe um tabuleiro e o exibe, chamando, alternadamente, as funções *draw_line* e *draw_tiles*;
- ***how_to_play***: exibe uma ajuda ensinando as regras do jogo e como jogar para o usuário, presente no arquivo “.how_to_play.txt”. É exibida uma linha (ou tabuleiro) de cada vez, podendo o usuário pressionar Enter para continuar lendo ou “E” para sair.

2.5. Módulo scoreboard

Módulo responsável por executar as ações referentes ao placar, o qual exibe apenas o melhor resultado (por tipo de jogo), baseado na pontuação. Nele há as funções:

- ***update_scoreboard***: recebe o tabuleiro que se deseja inserir (se possível) no placar, e verifica se sua pontuação é maior que a pontuação do jogo já armazenado no placar, que fica no arquivo “.score_*.txt”, em que o asterisco representa o tipo do jogo (“m” para médio, “h” para difícil, “p” para profissional e “c” para personalizado). Se sim, o tabuleiro, com suas informações, é armazenado no arquivo;
- ***scoreboard***: exibe, primeiramente, o menu de dificuldades, através da função *draw_diff*. Após o usuário selecionar a dificuldade do placar que deseja consultar, o arquivo do respectivo tipo é consultado e exibido, mostrando o tabuleiro e a pontuação daquele jogo.

2.6. Módulo run

Lida com as ações que ocorrem durante o jogo em si, como os possíveis movimentos do usuário. Possui as seguintes funções:

- ***prob***: recebe um inteiro n e retorna 0 ou 1, tendo uma chance em n de retornar 1;
- ***generate_number***: recebe um tabuleiro e gera um número nele. Cada casa vazia do tabuleiro tem a mesma chance de receber o número gerado, e isso é feito pelo uso da função *prob* e o valor *free* referente ao tabuleiro. Além disso, o valor gerado pode ser a base do tabuleiro N ou N^2 , sendo o primeiro N vezes mais provável de ocorrer;
- ***slide***: recebe o tabuleiro e a direção, representada por um inteiro, que o jogador escolheu e realiza o movimento correspondente para aquela direção. A função apenas calcula o movimento se a direção for para a esquerda. Se for para a direita, o tabuleiro é espelhado e a função é chamada com a direção para a esquerda. Se for para cima ou para baixo, o tabuleiro é transposto e a função é chamada para a direção correspondente (esquerda ou direita). A função ainda retorna 1 se alguma coisa foi alterada após o movimento ou 0 se nada mudou;
- ***move***: recebe o input do usuário, checando se é o comando de desfazer o último movimento (“U”), sair (“E”) ou um movimento, em que a função retorna um inteiro representando a direção do mesmo. Para os outros casos, são retornados inteiros que os codificam;

- **run**: Roda o jogo em si. Recebe como inteiros a base, o tamanho do tabuleiro, o tipo do jogo e um valor que é igual a 0 se é um novo jogo, ou 1 se é uma continuação de um jogo salvo. Para o caso de ser um novo jogo, a função verifica se há algum jogo salvo (no arquivo “.cont.txt”). Se sim, pergunta-se ao usuário se ele deseja gravar por cima desse jogo. Se sim, cria-se um novo tabuleiro através da função *new_board*. Se é continuação de um jogo salvo, o arquivo “.cont.txt”, que possui as informações do tabuleiro desse jogo, é lido. Em seguida a função entra num loop, que chama *clear_screen*, *draw_run_menu*, exibe a pontuação, chama *draw_board*, *move*, *slide*, *generate_number*, etc., checando condições como o jogador selecionar a opção de desfazer ou de sair. O loop é encerrado quando o jogador perde, e o tabuleiro é armazenado no arquivo “.cont.txt” para ser continuado posteriormente.

2.7. Módulo main

Função principal, administra o menu principal e suas funcionalidades. Só possui uma função, a função *main*, que chama *clear_screen*, *draw_menu*, e verifica a entrada do usuário. Se for “N”, chama a função *draw_diff*, e, de acordo com a dificuldade escolhida pelo usuário, chama a função *run*. No caso de ser “C”, chama a função *run* com a opção de continuar jogo salvo. Se for “S”, chama *scoreboard*, se for “H”, *how_to_play*, e, finalmente, se for “E”, termina o programa.