Universidade Federal de Ouro Preto – UFOP

Instituto de Ciências Exatas e Biológicas – ICEB
Departamento de Computação – DECOM
Disciplina: Introdução a Programação (BCC201)
Professores: Puca Huachi, Guilherme Augusto & Pablo Munhoz

Trabalho Prático: **2048**

O **2048** é um jogo de raciocínio criado em 2014, onde o objetivo é deslizar peças numeradas em um tabuleiro, combiná-las e criar um bloco com o número 2048.

Figura 1: Exemplo de uma etapa do jogo

O jogo

O **2048** clássico é jogado em um tabuleiro de 4×4, com peças numéricas que deslizam suavemente quando o jogador as move em um dos quatro sentidos disponíveis: para cima, para baixo, à esquerda e à direita.

A cada movimento, um novo número aparece aleatoriamente em um local vazio no tabuleiro (com um valor de 2 ou 4).

As peças (blocos) deslizam o mais longe possível na direção escolhida até que eles sejam interrompidos por qualquer outro bloco ou a borda do tabuleiro. Se duas peças do mesmo número colidem durante a movimentação, elas irão se fundir e a posição terá o valor total das duas peças que colidiram.

A peça resultante não pode se fundir com outra peça novamente na mesma jogada. Blocos com pontuação maior possuem cores diferentes.

Regras

O jogo começa com duas posições aleatórias do tabuleiro preenchidas.

A cada jogada, o jogador deve escolher uma direção (para cima, para baixo, para a esquerda ou para a direita).

Todas as peças se movem o máximo possível nessa direção, algumas se movem mais do que outras. Duas peças adjacentes (somente nessa direção) com números iguais se combinam em uma contendo a soma desses números.

Um movimento é válido quando pelo menos uma peça pode ser movida, inclusive por combinação.

Uma nova peça é gerada ao final de cada jogada em uma posição vazia escolhida aleatoriamente (se houver).

Na maioria das vezes, um novo **2** deve ser adicionado, mas ocasionalmente (10% das vezes), um **4**.

Para vencer, o jogador deve criar uma peça com o número 2048.

O jogador perde se não houver movimentos válidos possíveis.

O Trabalho Prático

Você deve implementar o jogo **2048** 2D para um único jogador em linguagem C para funcionar no terminal do linux.

Como não é exigido o funcionamento dos cliques de mouse, leiam os comandos do usuário pelo teclado. A interface do jogo é livre, porém o jogador deve ser capaz de entender o estado do jogo a cada jogada. Para o jogo, é necessário que seja feito um menu inicial com as seguintes opções:

- (R) Sair
  - Perguntar se deseja sair (Sim/Não). Se a resposta for não deve voltar ao menu.
- (N) Novo jogo
  - (4) jogo padrão 4×4.
  - (5) Jogo 5×5.
  - (6) Jogo 6×6.
- (J) Continuar o jogo atual
  - Continua o jogo de onde parou, caso já tenha começado um jogo. Caso contrário, continua no menu.
- (C) Carregar um jogo salvo
  - Carrega um jogo salvo em um arquivo.
- (S) Salvar o jogo atual
  - Salva o jogo atual em arquivo. O nome do arquivo deve ser solicitado.
- (M) Mostrar Ranking
  - Mostra o ranking salvo com as 10 melhores pontuações
- (A) Ajuda com as instruções de como jogar

Ao começar um jogo, o jogador deve digitar um dos comandos a seguir, até finalizar o jogo ou voltar para o menu inicial.

> **<a, d, s, w>:** Move as peças do tabuleiro para esquerda, direita, para baixo ou para cima, respectivamente.
>
> **<u>:** Desfazer o último movimento.

<t pos1, pos2>: Trocar duas peças de posição, ou seja, troca o conteúdo da posição **pos1** com o conteúdo da posição **pos2**.

voltar: Volta para o menu inicial.

Importante: seu programa deve proibir que o usuário execute comandos inválidos. O usuário deve ser alertado com uma mensagem de erro caso digite um valor inválido. Os menus e comandos podem aceitar letras maiúsculas ou minúsculas. Além disso, você não pode alterar os nomes dos comandos tão pouco os valores que devem ser digitadas nos menus.

Requisitos do jogo

Abertura das peças

A quantidade de peças inseridas a cada jogada, deve observar a seguinte regra:

- Jogo 4×4
  - Uma peça a cada rodada. O valor 2 terá 90% de chance e o valor 4 terá 10%.
- Jogo 5×5
  - Uma peça a cada rodada, O valor 2 terá 85% de chance e o valor 4 terá 15%.
- Jogo 6×6
  - Duas peças a cada rodada, O valor 2 terá 80% de chance e o valor 4 terá 20%.

Movimento "não-guloso"

As peças criadas pela combinação de outras peças não devem ser combinadas novamente durante a mesma jogada (movimento).

Ou seja, mover a fileira de peças de:

[2] [2] [2] [2]

para a direita deve resultar em:

[ ] [ ] [4] [4]

e não:

[ ] [ ] [ ] [8]

Prioridade da direção do movimento

Se mais de uma variante de combinação for possível, a direção do movimento indicará qual combinação terá efeito.

Por exemplo, mover a fileira de peças de:

[ ] [2] [2] [2]

para a direita deve resultar em:

[ ] [ ] [2] [4]

e não:

[ ] [ ] [4] [2]

Verificações durante o jogo

Verifique se há movimentos válidos. O jogador não deve conseguir ganhar uma nova peça tentando um movimento que não altere o tabuleiro.

Verifique se há uma condição de vitória e informe o jogador. Nesse caso, o jogador poderá continuar jogando ou voltar para menu principal.

Verifique se há uma condição de derrota e informe o jogador Nesse caso, o jogador poderá desfazer o último movimento, se ele possuir movimentos desfazer, ou voltar para menu principal.

Desfazer movimentos e trocar peças de posição

O jogador terá direito a desfazer o último movimento feito. A quantidade de chances de desfazer um movimento será definida pelo seu desempenho. Ele começa com zero chances de desfazer movimentos e a cada vez que conseguir uma peça com valor igual a 256, ele ganhará uma chance. O número de chances é acumulativo e todo vez que usar uma esse ela deve ser descontada.

O jogador também poderá trocar duas peças de posição. Assim como no movimento desfazer, ele começa sem nenhuma chance e toda vez que conseguir uma peça no valor 512 ele ganha uma possibilidade.

Para facilitar o comando de trocar as peças, o tabuleiro deve ser identificado. As linhas devem identificadas com letras e as colunas com números. Assim, em um tabuleiro 4×4 a primeira posição, no canto superior esquerdo, é a `A1` e a última, no canto inferior direito, é `D4`.

Pontuação

O jogador começa com a pontuação zerada e a cada movimento que ele conseguir juntar peças os valores das peças resultantes são somados a sua pontuação.

Começando um jogo

Quando o usuário selecionar a opção de novo jogo, deve ser solicitado o tamanho do tabuleiro e o seu nome. Em seguida o jogo deve iniciar com o placar, o número de trocas e o número de desfazer zerados.

Se a opção for de começar um jogo salvo deve ser solicitado o nome do arquivo e só depois continuar o jogo. Observe que no arquivo do jogo terá, todas as informações necessárias para continuar um jogo em andamento.

Características do programa

- A interface não precisa ser gráfica ou sofisticada, mas o jogador deve ser capaz de entender o estado do jogo a cada jogada.
- Para implementar o jogo você pode utilizar qualquer função da biblioteca padrão. É obrigatório o uso de `struct` e alocação dinâmica.
- O programa deve possuir um ranking com o nome dos jogadores com as 10 maiores pontuações. A ordem de classificação será em ordem decrescente. O programa deve armazenar o ranking em um arquivo.
- Não utilize variáveis globais no programa (Pode ser usado para definir constantes, se necessário). Faça o programa bem modular e com funções. A `main()` deve ser pequena e organizada. Pode-se utilizar passagem de parâmetros por valor ou referência nas funções.
- Coloque comentários no programa e utilize identificadores que definam com clareza a funcionalidade das funções e variáveis.
- Para selecionar as posições aleatórias das peças utilize a função `rand()` da biblioteca `stdlib.h`.

Lendo e salvando um jogo

O arquivo com as informações do jogo deve ser em formato texto com as informações armazenadas como segue:

Em breve!

Lendo e salvando o ranking

Um arquivo chamado ranking.dat deve ser gravado com as 10 maiores pontuações para cada tamanho de jogo. Segue o detalhamento do formato do arquivo:

Em breve!

Avaliação

O jogo deve ser implementado por meio de um programa em C. Inclua seu nome e número de matrícula como comentário em todos os arquivos .c e .h gerados.

Os seguintes itens será avaliados:

- Funcionamento adequado do programa.

- Atendimento ao enunciado do trabalho.
- Clareza do código (que deve ser devidamente comentado e identado).
- Utilização de funções e `struct`.
- Utilização de alocação dinâmica.
- Adequação da estrutura do programa (variáveis e comandos utilizados).
- Apresentação do trabalho.
- Compilação (códigos que não compilam serão zerados, e *warnings* diminuirão a nota). Utilizaremos o compilador GCC.
- [redacted]

Entrega

- Você deverá entregar o código fonte, se forem vários arquivos, compactar em formato ZIP
- A entrega deve ser feita pelo Moodle até as 3h00 do dia 26/08/2025.
- As entrevistas serão feitas, preferencialmente, durante os horários da aulas.

Exemplo de Execução

Você pode (e deve) customizar e melhorar as saídas do programa. A seguir segue um exemplo simples apenas para entendimento (os dados digitados pelo usuário estão destacados em azul):

Exemplo 1: Em breve!