

## Trabalho 1 — Campo Minado

### Instruções

O trabalho é em dupla (não pode ser individual) e pode ser submetido via SIGAA (arquivo compactado com todo o código-fonte) ou via GitHub Classroom.

### O Jogo

O jogo consiste em uma área matricial de  $n$  linhas por  $m$  colunas com um total de  $n*m$  células e  $k$  minas. Inicialmente, as células estão ocultas, ou seja, não é possível determinar se ela possui uma mina ou não, e o objetivo do jogo é revelar todas as células que não possuem minas. Assim, uma célula pode ser revelada por uma ação do usuário ou por uma das regras do jogo e isso pode resultar em uma **três situações possíveis**:

1. **Uma mina é revelada**: nesse caso jogo encerra com a derrota do usuário;
2. **Um número é revelado**: o valor indica a quantidade de minas adjacentes;
3. **Uma posição vazia é revelada**: nesse caso, o jogo revela automaticamente todas as células adjacentes a essa posição, visto que ela não possui minas adjacentes.

O usuário vence o jogo ao revelar todas as células que não contêm minas. Além disso, **o usuário pode marcar uma célula não revelada com uma bandeira caso ele acredite que não existe uma mina nessa posição**. Logo, **o usuário pode aplicar a ação revelar sobre um número desde que a quantidade de bandeiras adjacentes seja igual ao seu número**. Assim, **todas as células adjacentes que não estão marcadas por bandeiras são reveladas automaticamente**. Observe, que isso ocorrerá mesmo que as bandeiras estejam posicionadas incorretamente e assim revelando uma mina.

Por fim, a dificuldade do jogo irá determinar as dimensões da área do jogo, a quantidade de minas e se a primeira ação de revelar sempre será um número ou sempre será uma posição vazia.

## Interface do Jogo

A interface do jogo deve apresentar as seguintes ações:

1. Ação de revelar uma célula ou número;
2. Ação de marcar uma célula com uma bandeira;
3. Ação de verificar o tempo decorrido desde a primeira ação de revelar;

Além disso, usuário deve conseguir visualizar toda a área do jogo identificando quais células estão ocultas, marcadas por bandeiras, são vazias ou números como também a quantidade de minas presentes menos a quantidade de bandeiras marcadas. Também é necessário a apresentação de um feedback adequado ao usuário ao fim da partida indicando a vitória ou derrota.

A interface será textual (executada no terminal) e você é livre para determinar como usuário deve executar essas ações e como a área do jogo será apresentada.

## Detalhes de Implementação

Além das regras descritas, o programa deve registrar o tempo das partidas, isto é, tempo decorrido da primeira ação de revelar até o fim do jogo. No caso de vitória, o programa deve verificar se o tempo obtido é menor que um dos 10 últimos registros e nesse caso deve gravar o nome e o tempo do usuário. Só devem ser mantidos 10 registros para cada nível de dificuldade. O programa também deve apresentar três níveis de dificuldades:

1. **Iniciante**: a área do jogo possui 10 linhas e 10 colunas com um total de 10 minas;

2. **Intermediário**: a área possui 15 linhas por 15 colunas com um total de 40 minas e a primeira ação de revelar sempre revela uma posição vazia;
3. **Avançado**: a área possui 15 linhas por 30 colunas com 100 minas e a primeira ação de revelar sempre revela um número.

O programa ainda deve manter um registro da dificuldade atual, por exemplo, se a dificuldade for configurada para iniciante então todas as execuções subsequentes do jogo devem iniciar partidas no modo iniciante.

A dificuldade do jogo deve ser alterada por argumentos enviados ao programa via linha de comando, isto é, através dos parâmetros argc e argv. Além disso, o programa deve aceitar os seguintes argumentos:

- **-h ou --help**: apresenta um resumo das regras do jogo e dos argumentos;
- **-r ou --records**: apresenta os nomes dos usuários os tempos correspondentes dos melhores tempos para a dificuldade corrente;
- **-d <b | i | a> ou --difficulty <beginner | intermediary | advanced>**: altera a dificuldade do jogo e então inicia uma partida no modo iniciante (opção 'b' ou 'beginner'), intermediário (opção 'i' ou 'intermediary') ou avançado (opção 'a' ou 'advanced').

Se nenhum parâmetro for informado então uma nova partida deve ser iniciada na dificuldade configurada (a dificuldade iniciante é a padrão caso a dificuldade nunca tenha sido alterada).

## Avaliação

Para a avaliação parcial do projeto, deve ser implementada a interface do jogo e também o tratamento dos argumentos do programa.

A entrega final deve consistir da implementação de todas as funcionalidades descritas no documento, ou seja:

- Regras do jogo (principal critério);

- Persistência das informações;
- Tratamento dos argumentos;

Além disso, devem ser empregadas as funcionalidades da linguagem C++ em detrimento das correspondentes na linguagem C.

## **Guia de Implementação**

As seguintes perguntas podem ser respondidas para guiar o desenvolvimento do sistema:

1. Qual o fluxo principal de execução do programa?
2. Quais os fluxos alternativos de execução? Um diagrama pode auxiliar na visualização desses fluxos.
3. Quais as informações manipuladas em cada fluxo? Como elas são criadas e manipuladas?
4. Existe alguma relação entre as informações, isto é, elas podem ser agrupadas representando um mesmo conceito?
5. Como as informações serão representadas no programa?
6. Como as informações são transformadas, manipuladas e exibidas ao longo do programa?