

# Desafio de Fundamentos de Programação

## “Campo-minado”

Universidade Tecnológica Federal do Paraná (UTFPR), campus Apucarana  
Curso de Engenharia de Computação  
Disciplina de Fundamentos de Programação - FD61A  
Prof. Dr. Luiz Fernando Carvalho  
Prof. Dr. Rafael Gomes Mantovani

---

## 1 Objetivo

Campo minado é um popular jogo de computador para um jogador. Foi inventado por Robert Donner em 1989 e tem como objectivo revelar um campo de minas sem que alguma seja detonada. Este jogo tem sido reescrito para as mais diversas plataformas, sendo a sua versão mais popular a que vinha nativamente nas edições anteriores ao Windows 10 ???. O objetivo deste projeto é implementar um jogo de campo minado, sem interface gráfica, usando apenas os recursos de programação da linguagem C e realizando a interação com o jogador (usuário) via console.

### 1.1 Regras do jogo

A área de jogo consiste num campo de quadrados retangular (células). Na versão do jogo com interface gráfica, cada quadrado pode ter seu conteúdo revelado clicando sobre ele, e se o quadrado clicado contiver uma mina, então o jogo acaba. Se, por outro lado, o quadrado não contiver uma mina, uma de duas coisas poderá acontecer:

- um número aparece, indicando a quantidade de quadrados adjacentes que contêm minas; ou
- nenhum número aparece. Neste caso, o jogo revela automaticamente os quadrados que se encontram adjacentes ao quadrado vazio, já que não podem conter minas;

O jogo é ganho quando todos os quadrados que não têm minas são revelados.

Opcionalmente, o jogador pode marcar qualquer quadrado que acredita que contém uma mina com um caractere ‘B’, bastando indicar a posição da matriz que deseja deixar marcada (número da linha e coluna). Atentar apenas que a quantidade de quadrados marcados como bombas deve ser menor ou igual que a quantidade de bombas existentes no cenário simulado.

---

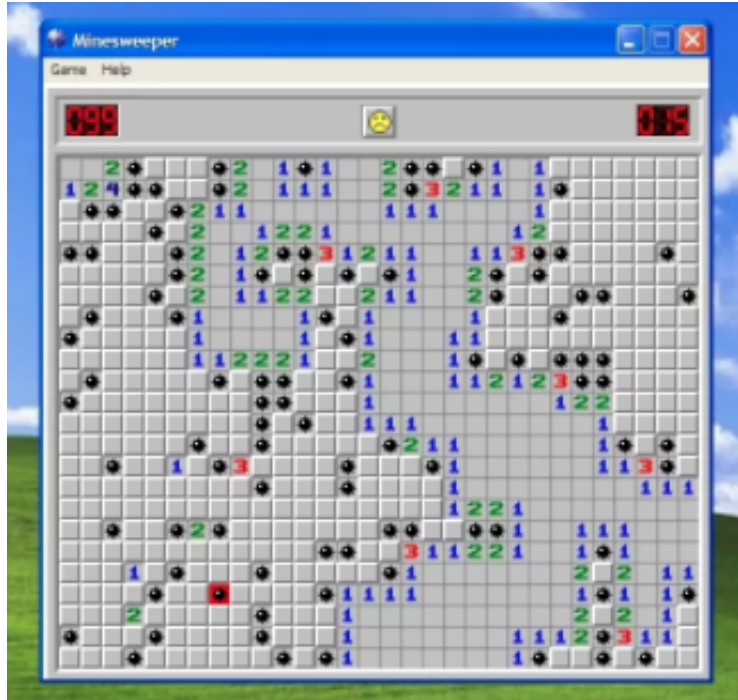


Figura 1: Tela do jogo Campo Minado no Windows XP.

## 2 Entradas

A entrada do programa é feita via arquivo texto. Ela consiste de:

- uma linha com dois inteiros M e N que correspondem ao número de linhas e de colunas do campo minado;
- um caractere ('F', 'M' ou 'D'), que indica o nível de dificuldade fácil, médio ou difícil, respectivamente. Assuma que as bombas devem ser posicionadas aleatoriamente no campo:
  - 'F' - 10% das células do campo minado correspondem a bombas;
  - 'M' - 20% das células do campo minado correspondem a bombas;
  - 'D' - 30% das células do campo minado correspondem a bombas.

## 3 Saída

Como saída, imprima no console, e também em um arquivo texto a ser gerado, o resultado do jogo.

Caso o usuário tenha vencido, ou seja, abriu todas as células sem ativar nenhuma das bombas, imprima na primeira linha a mensagem "Vitoria". Caso o usuário tenha perdido a partida, imprima a mensagem "Derrota". Em seguida, mostre:

- O campo minado resolvido:
  - bombas que não foram ativadas devem aparecer como caractere ‘0’;
  - a bomba que foi ativada em caso de derrota deve ser representada pelo caractere ‘X’;
  - números de bombas adjacentes em cada célula. Se esse número for zero, imprimir vazio (um espaço em branco).

**Dica:** Quando forem executar o programa com os arquivos texto é necessário manipular os argumentos **argc** e **argv** da função main. Para isso, deve-se executar o programa por linha de comando, obedecendo o seguinte padrão:

**<nome do programa> <arquivo de entrada> <arquivo de saída>**

Por exemplo, se o programa fonte se chamar “campominado.c”, o comando será:

**campominado entrada.txt saida.txt**

Dentro da função **main**, o valor de **argc** indica o número de parâmetros recebidos para a execução. Como são inseridos apenas os nomes dos arquivos de entrada e saída, nesse caso temos **argc = 3** (os dois arquivos mais o nome do programa). Além disso:

- **argv[0]** contém o nome do programa;
- **argv[1]** o nome do arquivo de entrada; e
- **argv[2]** o nome do arquivo de saída.

---

## 4 O que deve ser entregue?

Para entrega da atividade, o aluno deve enviar ao professor:

- a) todos os arquivos desenvolvidos em C para a resolução do problema;
  - b) relatório individual descrevendo a modelagem realizada, explicando todas as tomadas de decisão (estruturas de dados usadas, lógica desenvolvida, etc). Constar no relatório também casos de teste realizados (pode inserir figuras para mostrar as execuções realizadas);
  - c) apresentação oral para o professor em horário a ser combinado.
-

## 5 Orientações gerais

Além da funcionalidade desejada, implementar também o controle de erros, para lidar com exceções que possam ocorrer, como, por exemplo:

- problemas nas aberturas dos arquivos de entrada e saída;
- arquivo de entrada vazio (sem informação);
- arquivo de entrada fora do padrão esperado (dados faltantes, quantidade errada de linhas informadas, quantidade errada de palavras que se deseja buscar, etc.);
- etc.

Para acompanhamento do desenvolvimento, pode-se criar um repositório individual com o código desenvolvido na página pessoal do aluno no **github**. Ao elaborar seu programa, crie um único arquivo fonte (.c) seguindo o padrão de nome especificado:

**FDP-AT01-CampoMinado-<NomeAluno>.c**

Exemplo:

**FDP-AT01-CampoMinado-LuizFernandoCarvalho.c**

A entrega da atividade será via Moodle: o link será disponibilizado na página da disciplina. O prazo para a entrega é o final do semestre letivo corrente.

---

## 6 Links úteis

Números aleatórios em C:

- <http://linguagemc.com.br/valores-aleatorios-em-c-com-a-funcao-rand/>
- <https://www.ime.usp.br/~pf/algoritmos/aulas/random.html>
- <http://www.cplusplus.com/reference/cstdlib/rand/>

Arquivos em C:

- <https://www.inf.pucrs.br/~pinho/LaproI/Arquivos/Arquivos.htm>
- <https://www.geeksforgeeks.org/basics-file-handling-c/>
- <https://www.programiz.com/c-programming/c-file-input-output>

Argumentos de linha de comando (argc e argv):

- [https://www.tutorialspoint.com/cprogramming/c\\_command\\_line\\_arguments.htm](https://www.tutorialspoint.com/cprogramming/c_command_line_arguments.htm)
- <http://linguagemc.com.br/argumentos-em-linha-de-comando/>

- [http://www.univasf.edu.br/~marcelo.linder/arquivos\\_pc/aulas/aula19.pdf](http://www.univasf.edu.br/~marcelo.linder/arquivos_pc/aulas/aula19.pdf)
  - [http://www.inf.ufpr.br/cursos/ci067/Docs/NotasAula/notas-31\\_Argumentos\\_linha\\_comando.html](http://www.inf.ufpr.br/cursos/ci067/Docs/NotasAula/notas-31_Argumentos_linha_comando.html)
  - <http://www.dca.fee.unicamp.br/cursos/EA876/apostila/HTML/node145.html>
- 

## Referências

- [1] Wikipedia. Campo Minado. Disponível em: [https://pt.wikipedia.org/wiki/Campo\\_minado](https://pt.wikipedia.org/wiki/Campo_minado). Acesso em 25 de junho de 2021.
  - [2] Thomas H. Cormen,; Ronald Rivest; Charles E. Leiserson; Clifford Stein. Algoritmos - Teoria e Prática - 3ª Ed. Elsevier - Campus, 2012.
  - [3] Nivio Ziviani. Projeto de algoritmos com implementações: em Pascal e C. Pioneira, 1999.
  - [4] André Backes. Linguagem C: completa e descomplicada. Rio de Janeiro, RJ: Elsevier, 2013. 371 p.
  - [5] Luis Damas. Linguagem C. 10a edição. Editora LTC. 2007.
-