

3ª Prova de Algoritmos e Estruturas de Dados I

1º/07/2016

Perguntas comuns e suas respostas:

- P: Tenho uma dúvida na questão tal.
R: A compreensão do enunciado faz parte da prova.
- P: O que será avaliado?
R: A lógica, a criatividade, a sintaxe, o uso correto dos comandos, a correta declaração dos tipos, os nomes das variáveis, a indentação, uso equilibrado de comentários no código e, evidentemente, a clareza. A modularidade, correto uso de funções e procedimentos, incluindo passagem de parâmetros e bom uso de variáveis locais e globais serão especialmente observados.

QUESTÃO: Jogo *Batalha Naval* (100 pontos)

DESCRIÇÃO: O jogo de batalha naval é jogado em um tabuleiro retangular com N linhas e M colunas. Cada posição desse tabuleiro é um quadrado que pode conter água ou uma parte de um navio. Dizemos que dois quadrados são vizinhos se estes possuem um **lado em comum**. Se duas partes de navio estão em posições vizinhas, então essas duas partes pertencem ao mesmo navio. A regra do jogo proíbe que os quadrados de duas partes de navios distintos tenham um canto em comum (em outras palavras, que quadrados de duas partes de navios distintos compartilhem um vértice).

Cada disparo que um jogador faz deve ser feito em um dos quadrados do tabuleiro do outro jogador. Um jogador informa ao outro a coluna e a linha do quadrado alvo do disparo. Para que um navio seja destruído, o jogador deve acertar **todas as partes** desse navio. O jogador não pode atirar no mesmo lugar mais de uma vez.

TAREFA: Escreva um programa *Pascal estruturado* (ou seja, com procedimentos e/ou funções) para, dadas a configuração do tabuleiro e uma sequência de disparos feitos por um jogador, determinar o número de navios do outro jogador que foram destruídos.

ENTRADA: A primeira linha da entrada contém dois números inteiros N e M ($1 \leq N, M \leq 100$) representando respectivamente o número de linhas e de colunas do tabuleiro. As N linhas seguintes correspondem ao tabuleiro do jogo. Cada uma dessas linhas contém M caracteres, indicando o conteúdo da posição correspondente no tabuleiro. Se esse caractere for ".", essa posição contém água; se for "#", essa posição contém uma parte de um navio. A próxima linha contém um número K que é o número de disparos feitos pelo jogador ($1 \leq K \leq N \times M$). As K linhas seguintes indicam os disparos feitos pelo jogador. Cada linha contém dois inteiros L e C , indicando a linha e a coluna do disparo feito pelo outro jogador ($1 \leq L \leq N$ e $1 \leq C \leq M$).

SAÍDA: Uma única linha contendo um único número inteiro: **o número de navios destruídos**.

AVALIAÇÃO:

1. **70 pontos:** implementação na qual todos os navios são compostos por exatamente uma parte, ou seja, cada navio ocupa exatamente um quadrado (exemplo de entrada 1).
2. **30 pontos:** implementação na qual navios podem ocupar mais de um quadrado (exemplos de entrada 2 e 3); definições da implementação anterior podem ser reutilizadas.

Observe que não é estritamente necessário implementar ambas as versões, uma vez que a versão 2 também resolve a situação da versão 1.

EXEMPLOS:

Entrada 1 (6 navios):

```
5 5
.#.#
#....
...#.
#....
...#.
5
1 3
1 4
1 5
2 1
3 4
```

Saída:

4

Entrada 2 (4 navios):

```
5 5
.###
.....
#####
.....
#.##.
5
5 1
5 2
1 3
1 4
1 5
```

Saída:

2

Entrada 3 (3 navios):

```
7
.#....#
###.##
.#....#
....##
.#.##.
.####.
.....
8
1 1
1 2
2 1
2 2
2 3
3 2
5 2
6 2
```

Saída:

1