Universidade Federal de Viçosa
Departamento de Informática
INF 112 - Programação II
2º Semestre de 2021
Professor: Julio Cesar Soares dos Reis
Valor: 5 Pontos

Trabalho Prático 2

Objetivos

Consiste em rever conceitos básicos de programação bem como explorar técnicas como recursividade e *backtracking*.

Descrição

Um aluno da UFV se perdeu nas ruas de Viçosa após uma noite de festa. Ao chegar em casa, ainda meio desorientado, tal aluno decidiu fazer um programa, em C++, capaz de encontrar uma saída de um labirinto. Obviamente, a tarefa não foi realizada com sucesso e é portanto seu dever ajudar o estudante da UFV a não se sentir pior ainda. Sabe-se:

- o labirinto é representado por uma matriz, `maze`, de inteiros com n linhas e n colunas;
- a entrada do labirinto fica em `maze[1][0]`, a qual tem valor 0;
- a saída do labirinto pode ficar em qualquer posição da matriz. Essa posição terá valor 7;
- células da matriz podem ter valores 0, 1 ou 7. Zeros indicam uma passagem, ou seja, uma pessoa pode estar naquela posição, o valor 1 indica um obstáculo (parede) e o valor 7 (que ocorre uma única vez) indica que tal posição é uma saída;
- um viajante do labirinto que estiver na posição (i, j) pode se deslocar para $(i, j+1)$ ou $(i, j-1)$ ou $(i+1, j)$ ou $(i-1, j)$ desde que a nova posição escolhida não esteja ocupada com o valor 1 e seja uma posição válida dentro do labirinto.

Dada uma configuração de labirinto como a descrita acima, deseja-se saber se há uma saída ou não para o viajante (aluno UFV).

Abaixo, um exemplo em que a resposta é negativa:

```
1 1 1 0 0
0 0 0 0 1
1 0 1 1 1
1 1 7 1 0
1 1 1 1 1
```

Abaixo, um exemplo em que a resposta é positiva:

```
1 1 0 0 1
0 0 0 1 1
1 1 0 1 1
1 0 0 0 0
1 1 7 0 1
```

Dados n e a matriz `maze`, proponha um algoritmo baseado em *backtracking* com poda para decidir se o viajante (aluno da UFV) consegue ou não sair do labirinto.

Observação: fique atento para não deixar que o viajante entre em um ciclo dentro do labirinto.

Seu programa deve possuir construtor (leitura de `n` e alocação dinâmica da matriz `maze`), destrutor e procedimentos (ou funções) para:

1. ler a matriz `maze` (leitura dos dados) (`leMatriz`);
2. imprimir o labirinto (`imprimeLabirinto`).
3. verificar se há ou não uma saída do labirinto (`verificaSaidaLabirinto`);

O seu programa deve estar modularizado (.h e .cpp) e além disso, deve conter um `main.cpp`, que explore as funções implementadas.

Desenvolvimento e Entrega

O código fonte do programa deve ser desenvolvido em C++, estar bem identado e comentado. A entrega deve ser efetuada conforme agendado no PVANet Moodle. Para isso, você deve criar um projeto contendo os arquivos `.h`, `.cpp`, e `main.cpp` criados. Envie, através do PVANet Moodle, uma pasta compactada (.rar ou .zip) contendo o projeto. A pasta compactada deve conter informações do aluno (ex.: julio_reis-tp2.zip). Para correção, serão considerados os seguintes critérios:

1. Documentação (**1pt**).
   (a) Detalhamento do código.
   (b) Comentários, endentação.
2. Funcionamento correto (**2 pts**).
   (a) Compila e executa, não apresenta *crash*, etc.
3. Aplicação correta dos conceitos (**2 pts**).
   (a) Uso de ponteiros, gerenciamento de memória, explora adequadamente recursividade/*backtracking* com poda, etc.

Comentários Gerais

- Comece a fazer este trabalho logo: o prazo para terminá-lo está tão longe quanto jamais poderá estar! :)
- O trabalho é individual (grupo de UM aluno);
- Trabalhos copiados serão penalizados (NOTA Zero).