

Trabalho 3 — Encontre a Saída

Introdução à Ciência de Computação

Prof. Moacir Ponti / João Batista

Monitores/PAE: Gabriel Cruz, Alex Sander Silva, Mariana Rodrigues,
Cezar Abadio Filho, Kelvin Oliveira

Implemente suas atividades sem compartilhar ou olhar código de seus colegas. Procure usar todos os conceitos vistos nas aulas. Documente a sua aplicação por meio de comentários no programa.

*“So, the Labyrinth is a piece of cake, is it?
Well, let’s see how you deal with this little slice...”*

Descrição do trabalho

Esse trabalho tem como objetivo treinar o uso de estruturas de reetição e arranjos, fazendo o aluno pensar em:

1. quais valores serão armazenados na memória;
2. como os arranjos (vetores, matrizes, etc.) são organizados na memória;
3. como processar os dados para obter o resultado desejado.
4. como modularizar o código via uso de funções.

Tarefa

Neste trabalho, será implementado um programa para andar em um labirinto e encontrar sua única saída. O programa deverá ler a posição inicial e o labirinto em si e caminhar pelo labirinto até descobrir sua saída. As seguintes regras se aplicam:

- O labirinto possui tamanho fixo e representado por uma matriz 7x7 em um total de 49 “casas”, conforme a Figura 1;
- A presença ou não de obstáculos é dada por uma matriz de inteiros onde o 0 representa caminho livre e o 1 representa presença de obstáculos. A Figura 2 ilustra a representação de um labirinto por uma matriz de inteiros 0 e 1;
- O labirinto deverá **obrigatoriamente** ser lido pelo programa por uma função;
- Só existem duas casas livres na extremidade da matriz: uma delas, definida no início do programa, é a entrada do labirinto; a segunda extremidade livre é obrigatoriamente a saída.

- O caminho até a saída não ultrapassará 20 casas;
- O caminho até a saída deve ser impresso de acordo com o número da casa da matriz, conforme Figura 1;
- O caminho impresso deverá ser o caminho mais curto até a saída, sem incluir repetições de casa ou tentativas frustradas do programa que resultem em becos sem saída;
- Não é permitido se locomover em diagonais.

0	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	32	33	34
35	36	37	38	39	40	41
42	43	44	45	46	47	48

Figura 1: Matriz que representa o labirinto.

					5
	8		10	11	12
	15		17		
	22	23	24	25	26
		30			33
	36	37	38		40
	43				

(a) Labirinto

1	1	1	1	1	0	1
1	0	1	0	0	0	1
1	0	1	0	1	1	1
1	0	0	0	0	0	1
1	1	0	1	1	0	1
1	0	0	0	1	0	1
1	0	1	1	1	1	1

(b) Matriz de inteiros

Figura 2: Representação de um labirinto em uma matriz de inteiros.

Dicas

Tenha em mente que você estará em um beco sem saída se o único caminho que puder seguir já foi percorrido. Para fazer essa verificação, é necessário armazenar o caminho já realizado em algum lugar! :)

Para facilitar a execução, tente encontrar uma relação matemática entre o par (linha,coluna) e o número da casa da matriz.

Entrada

A entrada será composta por (não será necessário realizar a verificação das entradas):

- 49 inteiros de valores 0 ou 1 que representarão, respectivamente, caminho livre e presença de obstáculo de cada casa do labirinto, **conforme a ordem das casas dada pela Figura 1** (o primeiro número representa a casa 0, o segundo a casa 1 e assim por diante até a casa 48).
- 1 inteiro de valor 0 a 48 que representa o número da casa da posição inicial.

A entrada abaixo representa a entrada do labirinto da Figura 2 (A matriz é dada em uma única linha).

```
1 1 1 1 1 0 1 1 0 1 0 0 0 1 1 0 1 0 1 1 1 1 0 0 0 0 0 1 1 1 0 1 1 0 1 1 0 0 0 1 0 1 1 0 1 1 1 1 1
43
```

Saída

O programa deve imprimir as casas percorridas até a saída, **incluindo as casas de entrada e saída, e com um espaço (' ') entre elas e terminada por uma quebra de linha** (caracter `\n`). Por exemplo, para o labirinto da Figura 2, a saída deverá ser:

```
43 36 37 30 23 24 17 10 11 12 5 \n
```

Notar que há um espaço entre a casa de saída e o caracter `\n`.

Instruções

O trabalho será avaliado levando em consideração:

1. Realização dos objetivos / lógica utilizada
2. Uso de comentários e estrutura no código (e.g. indentação, legibilidade, modularização)
3. Resultado da plataforma run.codes
4. Bom uso da memória

ATENÇÃO:

- O projeto deverá ser entregue apenas pelo (<http://run.codes>) no formato de **código fonte**, ou seja apenas o código C.
- O prazo está no sistema run.codes
- Em caso de projetos **copiados** de colegas ou da Internet, todos os envolvidos recebem nota zero. Inclui no plágio a cópia com pequenas modificações, cópia de apenas uma parte ou função. Portanto programe seu próprio trabalho.