

Universidade Federal do Estado do Rio de Janeiro - UNIRIO
Bacharelado em Sistemas de Informação
Estrutura de Dados 2
Pedro Moura



Trabalho Final

Raquel Godoy Thiele

12 de dezembro de 2017

Sumário

Introdução	2
Mapeamento do Labirinto em um Grafo	3
Classes em Java	4
Encontrando a saída	5

1.Introdução

Foi proposto que neste trabalho se fizesse um código-fonte em Java. Neste código deveria se resolver o seguinte problema: dado o labirinto (Figura 1), modelar este em um grafo de modo que cada encontro ou “interseção” de caminhos seja um vértice e cada possível caminho entre 2 encontros seja uma aresta.

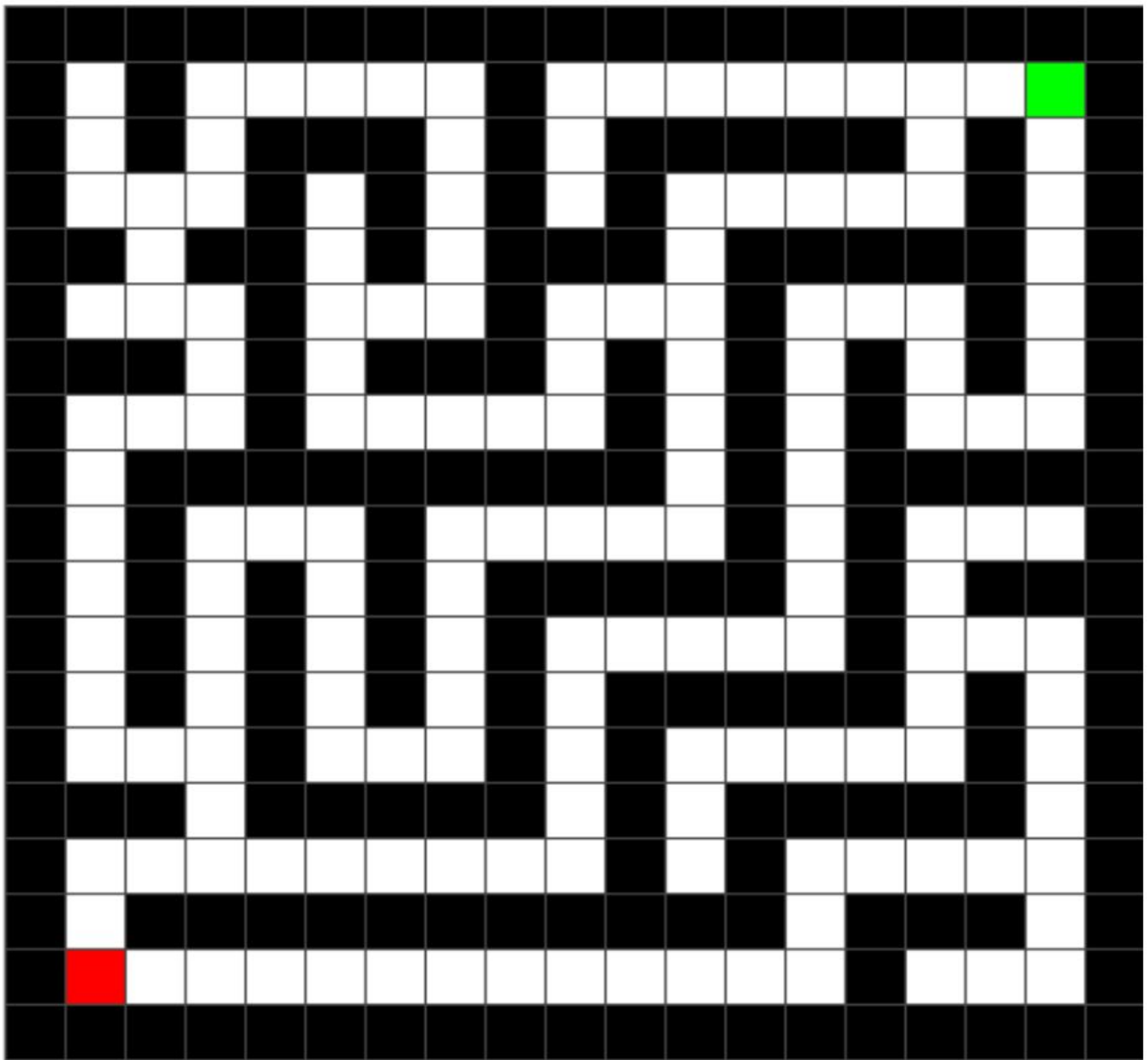


Figura 1: Labirinto

Posteriormente, deve-se desenvolver um código que receba este grafo (o labirinto) e encontre um caminho entre a entrada, quadrado verde, e a saída, quadrado vermelho.

2.Mapeamento do Labirinto em um Grafo

O labirinto foi mapeado como mostra a Figura 2.

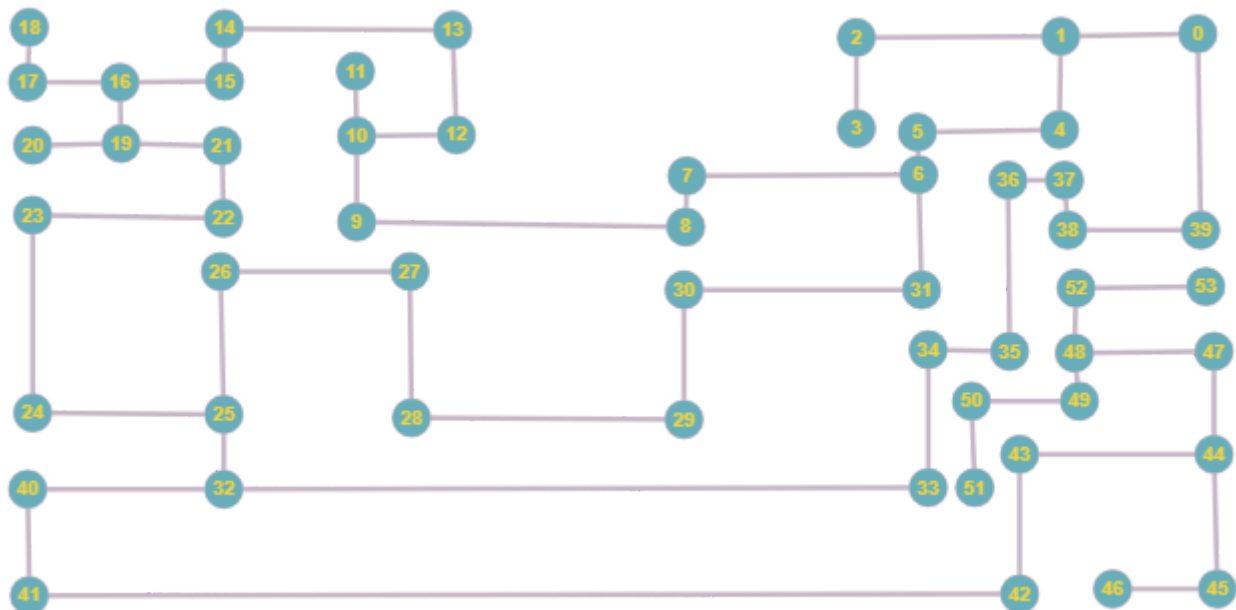


Figura 2: Grafo do Labirinto

Este grafo contém 54 vértices e 55 arestas. A lista de arestas deste grafo é:

0 1	12 13	25 32	40 41
0 39	13 14	26 27	41 42
1 2	14 15	27 28	42 43
1 4	15 16	28 29	43 44
2 3	16 17	29 30	44 45
4 5	16 19	30 31	44 47
5 6	17 18	32 33	45 46
6 7	19 20	32 40	47 48
6 31	19 21	33 34	48 49
7 8	21 22	34 35	48 52
8 9	22 23	35 36	49 50
9 10	23 24	36 37	50 51
10 11	24 25	37 38	52 53
10 12	25 26	38 39	

3. Classes em Java

As classes em Java usadas neste trabalho foram retiradas do código-fonte disponibilizado em "<https://algs4.cs.princeton.edu/40graphs/>" que fazem parte do livro *Algorithms, 4th Edition* por Robert Sedgewick e Kevin Wayne.

As classes usadas foram:

- Bag: classe auxiliar que representa um conjunto (multiset) de objetos genéricos, na forma de uma lista encadeada.
- Depth First Paths: única classe alterada para a realização deste trabalho. Ela possui a implementação do algoritmo que encontra todos os caminhos partindo de um vértice para todos os outros em um grafo não direcionado.
- Graph: classe auxiliar que representa o grafo não direcionado com vértices que vão de 0 ao número de vértices -1.
- In: classe auxiliar que representa um input, fornecendo métodos para ler strings e números a partir de uma entrada, um arquivo, uma URL ou um soquete.
- StdIn: classe auxiliar que fornece métodos estáticos para ler strings e números a partir da entrada.
- StdOut: classe auxiliar que fornece métodos para imprimir strings e números na saída.

4. Encontrando a saída

Com o auxílio da classe `DepthFirstPaths`, o caminho da origem, aqui mapeado como vértice 0 até a saída, mapeado como vértice 41, foi: 0-39-38-37-36-35-34-33-32-40-41, como mostra a Figura 3.



Figura 3: Labirinto com o caminho encontrado