

Python 3 Days Challenge  
(P3DC)  
*An Amazing Problem to Solve*  
— EPITA —

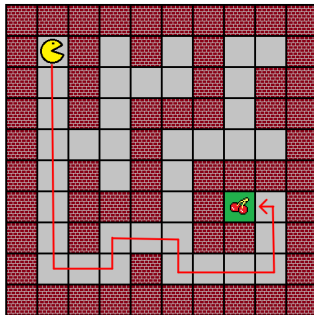
M. Angoustures & R. Dehak & R. Erra  
& M. Jivalian & A. Letois

April 2020

## Your main goal :

You have to solve a maze.

- You start from a fixed position in the maze.
- You need to find the fastest way to reach the fixed end point.



Example of a maze solved

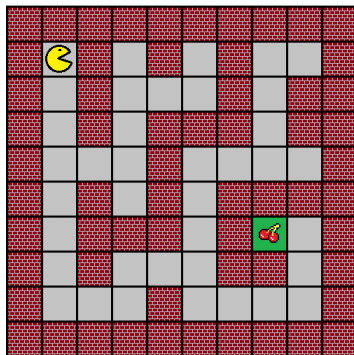
What do you have as input?

You have a matrix which describe every element of the maze

1	1	1	1	1	1	1	1	1	1
1	s	1	0	1	0	1	0	0	1
1	0	1	0	0	0	1	0	1	1
1	0	1	0	1	1	1	0	1	1
1	0	0	0	1	0	0	0	0	1
1	0	1	0	1	0	1	1	1	1
1	0	1	1	1	0	1	e	0	1
1	0	1	0	1	0	1	1	0	1
1	0	0	0	1	0	0	0	0	1
1	1	1	1	1	1	1	1	1	1

Matrix format of the maze

=



Picture of the respective maze

## Matrix description

Main element of the matrix has a rule in maze

Rules :

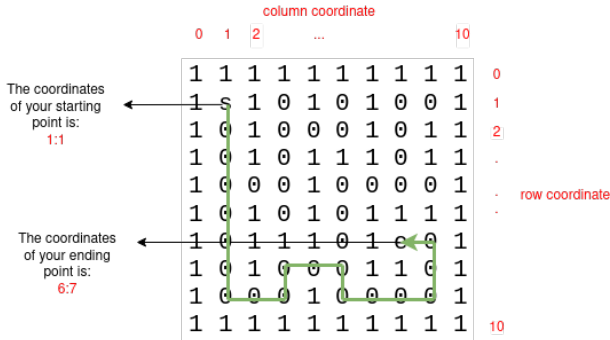
- **0 : path**  
you have to move this  
block
- **1 : wall**  
you cannot move to this  
block
- **s : start**  
your starting point
- **e : end**  
your ending point

1	1	1	1	1	1	1	1	1	1
1	s	1	0	1	0	1	0	0	1
1	0	1	0	0	0	1	0	1	1
1	0	1	0	1	1	1	0	1	1
1	0	0	0	1	0	0	0	0	1
1	0	1	0	1	0	1	1	1	1
1	0	1	1	1	0	1	e	0	1
1	0	1	0	1	0	1	1	0	1
1	0	0	0	1	0	0	0	0	1
1	1	1	1	1	1	1	1	1	1

Example

## What do we expect as a result ?

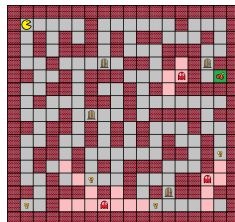
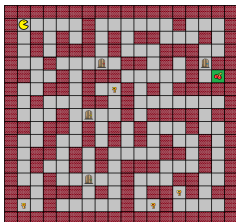
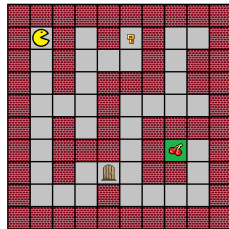
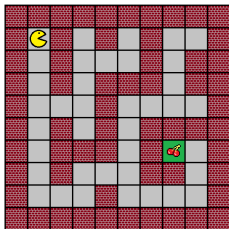
You need to provide a **list of tuples** which describe :  
the matrix coordinate point of the path from start to the end



**Your result:**

```
[(1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (6, 1),  
(7, 1), (8, 1), (8, 2), (8, 3), (7, 3),  
(7, 4), (7, 5), (8, 5), (8, 6), (8, 7),  
(8, 8), (7, 8), (6, 8), (6, 7)]
```

## 4 types of maze to solve : from easiest to hardest



## Different obstacles to overcome








image	matrix notation	description
	g	A red door prevent to to move forward until you have the red key to open it.
	f	A red key which let you to open the red door
	c	A green door prevent to to move forward until you have the green key to open it.
	d	A green key which let you to open the green door

image	matrix notation	description
	i	A blue door prevent to to move forward until you have the blue key to open it.
	h	A blue key which let you to open the blue door
	2 or more	The ghost has a range of 2 cells or more (pink in the pictures exemples) which kill you in all direction if you move on it. Avoid it!!!



## Your final result

Program a solver to compute the fastest path from start to the end by :

- Finding the right color key for right color door
- Avoiding the ghost's line of sight.

If you find different paths, return the shortest. Don't forget : you always must return a **unique** path, i.e. a unique list of tuples.

## Expected usage and result example

**python solvemaze.py maze1.txt** will give us :

```
[(1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (6, 1), (7, 1), (8, 1), (8, 2), (8, 3), (7, 3), (7, 4), (7, 5), (8, 5), (8, 6), (8, 7), (8, 8), (7, 8), (6, 8), (6, 7)]
```

## Warning

- DO NOT HARD CODE THE PATH!!!
- You will have new much bigger matrix matching completley new maze. Your program will have to find each path for each matrix.

## Your secondary goal :

You have to program the game interface.

- You could use Pygame or Pygame zero.
- You need to animate the moving of the pacman to the end.

We will provide you the images matching the element of the maze.

## Where to sent your work

Sent <sup>a</sup> to p3dcmssc@protonmail.com

- 1 Your python code
- 2 **and** your Jupyter Notebook if you have one.
- 3 and (mandatory) : add **all** your Family and First names both in your python code and in your Jupyter Notebook if you have one.
- 4 and (mandatory) : add YOUR TEAM NUMBER in your python file and in your Jupyter Notebook if you have one.
- 5 and (mandatory) : add in your email (object) : <Python Week 3 Days Challenge : <Team number > >
- 6 A unique email please !

---

a. Don't use podcmssc@protonmail.com !

Get the best score you can !

... **Congratulations and *bon courage* to all of you.**