

Extreme Lab #1 - 마방진 (Magic Square)

Copyright 2015 © document created by TeamLab.Gachon@gmail.com
(<mailto:TeamLab.Gachon@gmail.com>)

Introduction

처음으로 나오는 Extreme Homework. 이때까지 모든 Lab은 모두가 할 수 있게 설계되었다면 이번 숙제는 정말 열심히 따라온 학생들만 풀수 있는 어려운 Lab이다.
이번 Lab은 마방진 게임이다. 드라마 “뿌리깊은 나무” 에서 어린 세종으로 나왔던 송중기가 시간을 보내기 위해 열심히 풀었던 문제이기도 하다



마방진 문제는 의외로 간단하다. 그러나 로직을 for문과 if문으로 표현하는데 익숙하지 않다면 표현하는데 상당한 어려움을 겪게 될 것이다.

숙제 template 파일 다운로드

먼저 숙제 template 파일을 cs50 서버로 부터 다운로드 받는다. 로그인 후 나타나는 bash shell 에서 다음과 같은 명령을 입력하자.

```
python3.4 submit_assignment.py -get magic_square
```

정상적으로 다운했다면 현재 디렉토리에 `magic_square.py` 파일 생성되었을 것이다. `ls` 명령어로 확인하자.

baseball_game.py 파일 Overview

`vim` editor 로 `magic_square.py` 을 열어 전체적인 개요를 보자. `vi magic_square.py` 명령으로 파일을 열어보면 `main` 함수와 여러개의 함수들이 존재할 것이다. 각 함수들은 여러분께서 직접 작성해서 제출해야 하는 함수이고, `main` 함수는 실제 마방진 프로그램을 실행하는 함수이다. 각 함수의 구현 내용은 아래와 같다.

| 함수 | 설명 |
|-------------------------------------|--|
| <code>get_zero_matrix</code> | 정수형(integer)값을 N을 입력받아, N by N 정사각 행렬 형태인 two dimensional list를 반환함. list내 모든 element의 값은 0으로 초기화되어 있음 |
| <code>is_validate_number</code> | 문자열(string) 값을 입력받아, 값이 정수형 문자열이고 3보다 크고 20보다 작을 경우에는 True 그렇지 않을 경우에는 False를 반환함 |
| <code>is_4even_number</code> | 정수형(integer)값을 N을 입력받아, N이 4의 배수이면 True 그렇지 않으면 False를 반환함 |
| <code>is_odd_number</code> | 정수형(integer)값을 N을 입력받아, N이 홀수이면 True 그렇지 않으면 False를 반환함 |
| <code>get_4even_magic_square</code> | 4의 배수인 정수형(integer) 문자열(string)값 N을 입력받아, 마방진으로 구성된 N by N 정사각 행렬을 반환함. 반환되는 정사각행렬은 two dimensional list로 되어있음 |
| <code>get_odd_magic_square</code> | 홀수인 정수형(integer) 문자열(string)값 N을 입력받아, 마방진으로 구성된 N by N 정사각 행렬을 반환함. 반환되는 정사각행렬은 two dimensional list로 되어있음 |
| <code>is_magic_sqaure</code> | 정사각행렬 형태의 two dimensional list를 입력받아, 입력받은 list가 마방진인지 아닌 확인함 |

마방진 이해하기

마방진은

- $n \times n$ 개의 수를 가로, 세로, 대각선 방향의 수를 더하면 모두 같은 값이 나오도록 $n \times n$ 행렬에 배열한 것
- 일반적으로 마방진의 각 칸에는 1부터 $n \times n$ 까지의 수가 한 개씩 들어감. n 이 2일 때를 제외하고 항상 존재

간단한 3 by 3 행렬의 마방진은 다음 예제와 같다 (메모리스트의 상상 노트
(<http://memorist.tistory.com/151>)).

| | | |
|---|---|---|
| 4 | 9 | 2 |
| 3 | 5 | 7 |
| 8 | 1 | 6 |

가로, 세로, 대각선 모든 합이 15

$$4 + 9 + 2 = 15$$

$$3 + 5 + 7 = 15$$

$$8 + 1 + 6 = 15$$

$$4 + 3 + 8 = 15$$

$$9 + 5 + 1 = 15$$

$$2 + 7 + 6 = 15$$

$$4 + 5 + 6 = 15$$

$$2 + 5 + 8 = 15$$

이때 홀수 마방진과 4배수 짝수 마방진의 구성하는 방법은 아래와 같다.

홀수 마방진 구성 방법

1. 정사각형의 맨 아랫줄 가운데에 숫자 1 을 둔다.
2. 이전 숫자 위치에서 오른쪽 아래칸이 비어 있으면 다음 숫자를 채운다.
3. 이전 숫자 위치에서 오른쪽 아래칸이 채워져 있으면 이전 숫자의 위칸에 다음 숫자를 채운다.
4. 오른쪽 아래칸이 사각형의 영역 밖이면 다음의 규칙을 따른다.
 - 4-1. 수평 및 수직으로 이동해서 마지막 칸이 비어 있으면 해당 칸에 숫자를 채운다.
 - 4-2. 수평 및 수직으로 이동해도 칸이 없는 경우 이전의 숫자 위치 위쪽 칸에 다음 숫자를 채운다.

4배수 짝수 마방진 구성 방법

1. 대각선의 위치만 1 부터 시작해서 해당칸이 몇 번째 칸인지 숫자를 채운다.
2. 맨 오른쪽 아래부터 위로 올라오면서 채워지지 않은 숫자를 순서대로 채운다.

상당히 복잡해 보이지만 사실 인터넷에 코드는 많다. 찾아보고 해도 그 누구도 뭐라하지 않는다. 완성이 중요하다.

main 함수 수정하기

위의 함수도 상당히 어렵지만 본 Lab의 가장 어려운 숙제는 바로 main 함수를 수정하는 일이다. main 함수의 기본 template은 아래와 같다.

```
def main():
    user_input = 999
    print("Play Magic Square Game!!")
    # ===Modify codes below=====

    # =====
    print("Thank you for using this program")
    print("End of the Game")
```

메인 함수는 다음과 같은 규칙이 있다.

입력검사

1. 입력된 값이 문자열이나 소수점이 포함된 숫자일 경우 “Wrong Input! Input Again Please” 출력된다
2. 입력된 값이 3보다 작거나 20보다 클 경우 “Wrong Input! Input Again Please” 출력된다
3. 입력된 값이 홀수 또는 4배수 값이 아닐 경우에는 “Wrong Input, Only Input odd number or 4n number” 출력된다.

사용자가 3~20 사이의 정수를 정확히 입력했을 경우 아래와 같이 출력된다.

출력 결과

1. Matrix 형태로 출력하되 출력되는 글자를 포함하여 5칸이 사용된다. 이때 사용하는 문법은 "{:5d}".format(10) 형태로 활용하면 위치를 조정하여 정수값을 출력할 수 있다.
2. 마방진이 출력된후 각 줄의 합을 “The sum of each row is 15” 와 같은 형태로 출력한다.
3. 전체 마방진의 값을 “The sum of the matrix is 45” 형태로 출력한다.
4. 마지막 한줄은 띄워쓰기를 한다.

실제로 작성된 프로그램의 실행화면은 다음과 같다.

```
estella.gachon.ac.kr - PuTTY

Play Magic Square Game!!
Input the size of magic square: 3
The Magic Matrix of size 3 x 3 is:
  8   1   6
  3   5   7
  4   9   2
The sum of each row is 15
The sum of the matrix is 45

Input the size of magic square: 4
The Magic Matrix of size 4 x 4 is:
  1  15  14   4
 12   6   7   9
  8  10  11   5
 13   3   2  16
The sum of each row is 34
The sum of the matrix is 136

Input the size of magic square: 6
Wrong Input, Only Input odd number or 4n number
Input the size of magic square: a
Wrong Input! Input Again Please
Input the size of magic square: b
Wrong Input! Input Again Please
Input the size of magic square: 32.3
Wrong Input! Input Again Please
Input the size of magic square: 21
Wrong Input! Input Again Please
Input the size of magic square: 5
The Magic Matrix of size 5 x 5 is:
 17  24   1   8  15
 23   5   7  14  16
  4   6  13  20  22
 10  12  19  21   3
 11  18  25   2   9
The sum of each row is 65
The sum of the matrix is 325

Input the size of magic square: 0
Thank you for using this program
End of the Game
```

숙제 제출하기

모든 함수를 다 수정했다면, 아래와 같이 제출하자

```
python3.4 submit_assignment.py -submit gowithflow.py
```

제대로 작성했다면 아래와 같은 메시지가 뜰 것이다.

| ----- | ----- | ----- |
|------------------------|---------|----------|
| Function Name | Passed? | Feedback |
| ----- | ----- | ----- |
| get_4even_magic_square | PASS | Good Job |
| get_zero_matrix | PASS | Good Job |
| is_odd_number | PASS | Good Job |
| is_validate_number | PASS | Good Job |
| is_4even_number | PASS | Good Job |
| is_magic_sqaure | PASS | Good Job |
| main | PASS | Good Job |
| get_odd_magic_square | PASS | Good Job |
| ----- | ----- | ----- |

Next Work

Extreme Lab은 사실 A+를 받기 위한 필수 관문이다. 아무나 쉽게할 수 있는 일이 아니다. 일종의 프밍 클래스의 명예라고나 할까? 아무나 할 수 있는 일은 아니니 자랑스럽게 생각하길 바란다. 수고했다.

Human knowledge belongs to the world - from movie 'Password' -

Footnotes