

# EE231002 Introduction to Programming

## Lab06. Magic Squares

**Due: Apr. 5, 2014**

A magic square is an  $N \times N$  matrix that fills with integers from 1 to  $N^2$  and satisfies the following conditions.

1. All the row sums are equal,
2. All the column sums are equal,
3. The sums of the two diagonals are equal to the row sum and column sum.

For example, two  $3 \times 3$  magic squares are shown below.

2	7	6
9	5	1
4	3	8

2	9	4
7	5	3
6	1	8

Note that all the row sums, column sums and the two diagonal sums are equal to 15. In fact, given an  $N \times N$  magic square, it is easy to show that the row sums, column sums and diagonal sums are all equal to  $\frac{N(N^2 + 1)}{2}$ .

In this lab, you will write a **C** program to find **all**  $N \times N$  magic squares using exhaustive search, where  $N = 3, 4, 5$ . In your program, please define a macro for  $N$  as the following.

```
#define N 3
```

In this way, we can change  $N$  easily to test the program. Of course, each program execution finds only one set of magic squares. A different set can only be generated after editing (**vim**) and recompilation. Example of program output for  $N = 3$  case is shown below.

---

```
$ ./a.out
```

```
Magic Square #1:
```

```
 2  7  6
 9  5  1
 4  3  8
```

```
Magic Square #2:
```

```
 2  9  4
 7  5  3
 6  1  8
```

```
....
```

```
Magic Square #8:
```

```
 8  3  4
 1  5  9
 6  7  2
```

```
Total number of magic squares found = 8
```

---

### Notes.

1. Create a directory **lab06** and use it as the working directory.
2. Name your program source file as **lab06.c**.
3. The first few lines of your program should be comments as the following.

```
/* EE231002 Lab06. Magic Squares
   ID, Name
   Date:
*/
```

4. After finishing editing your source file, you can execute the following command to compile it,

```
$ gcc lab06.c
```

If no compilation errors, the executable file, **a.out**, should be generated, and you can execute it by typing

```
$ ./a.out
```

5. After you finish verifying your program, you can submit your source code by

```
$ ~ee231002/bin/submit lab06 lab06.c
```

If you see a "submitted successfully" message, then you are done. In case you want to check which file and at what time you submitted your labs, you can type in the following command:

```
$ ~ee231002/bin/subrec
```

It will show the last few submission records.

6. You should try to write the program as efficient as possible. The format of your program should be compact and easy to understand. These are part of the grading criteria.