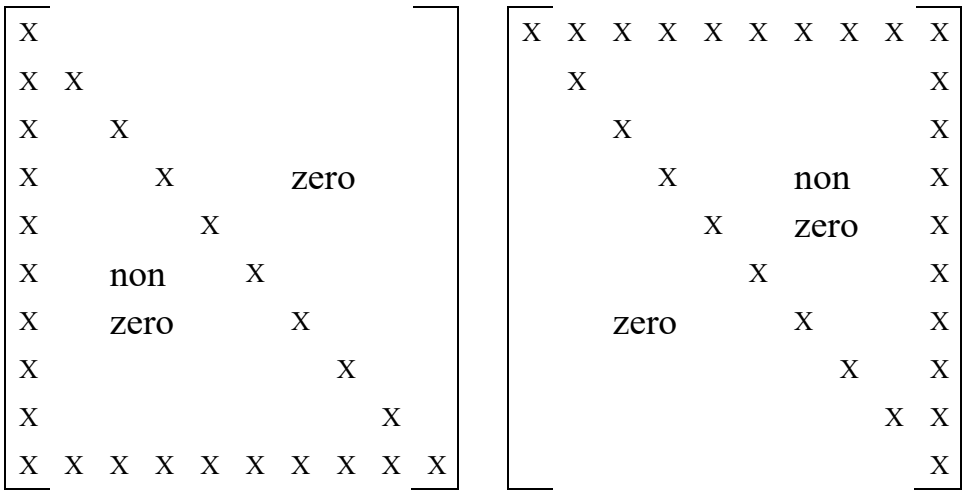
Data Structure Assignment 2

Paper Homework

(textbook p. 99)

3. A $triangular\ matrix$ is one in which either all the elements above the main diagonal or all the elements below the main diagonal of a square matrix are zero. Figure below shows a lower and an upper triangular matrix. In a lower triangular matrix, a, with n rows, the maximum number of nonzero terms in row *i* is *i* +1.



Thus, the total number of nonzero terms is

$$\mathrm{d} = \sum_{i=0}^{n-1} (i+1) = n(n+1)/2.$$

Since storing a triangular matrix as a two dimensional array wastes space, we would like to find a way to store only the nonzero terms in the triangular matrix. Find an addressing formula for the elements a_{ij} so that they can be stored by rows in an array $b[n(n+1)/2-1]$, with $a[0][0]$ being stored in $b[0]$.

General Information:

- Deadline : 2018/10/24(Please submit to TA after class)
- Late homework will not be accepted.
- Please write on **A4** papers.
- Notice : You won't get any point if you only write the answer, please list your process and reason.
- Any copies will be scored as zero. Do not plagiarize

Programming Homework

(textbook p. 102)

9. There are a number of problems, known collectively as “random walk” problems, that have been of longstanding interest to the mathematical community. All but the most simple of these are extremely difficult to solve, and, for the most part, they remain largely unsolved. One such problem may be stated as:

A (drunken) cockroach is placed on a given square in the middle of a tile floor in a rectangular room of size $n \times m$ tiles. The bug wanders (possibly in search of an aspirin) randomly from tile to tile throughout the room. Assuming that he may move from his present tile to any of the eight tiles surrounding him (unless he is against a wall) with equal probability, how long will it take him to touch every tile on the floor at least once?

Hard as this problem may be to solve by pure probability techniques, it is quite easy to solve using a computer. The technique for doing so is called "simulation." This technique is widely used in industry to predict traffic flow, inventory control, and so forth. The problem may be simulated using the following method:

An $n \times m$ array $count$ is used to represent the number of times our cockroach has reached each tile on the floor. All the cells of this array are initialized to zero. The position of the bug on the floor is represented by the coordinates $(ibug, jbug)$. The eight possible moves of the bug are represented by the tiles located at $(ibug + imove[k], jbug + jmove[k])$, where $0 \leq k \leq 7$, and

$$\begin{aligned} imove[0] &= -1 & jmove[0] &= 1 \\ imove[1] &= 0 & jmove[1] &= 1 \\ imove[2] &= 1 & jmove[2] &= 1 \\ imove[3] &= 1 & jmove[3] &= 0 \\ imove[4] &= 1 & jmove[4] &= -1 \\ imove[5] &= 0 & jmove[5] &= -1 \\ imove[6] &= -1 & jmove[6] &= -1 \\ imove[7] &= -1 & jmove[7] &= 0 \end{aligned}$$

A random walk to any one of the eight neighbor squares is simulated by generating a random value for k, lying between 0 and 7. Of course, the bug cannot move outside the room, so that coordinates that lead up a wall must be ignored, and a new random combination formed. Each time a square is entered, the count for that square is incremented so that a nonzero entry shows the number of times the bug has landed

on that square. When every square has been entered at least once, the experiment is complete.

Write a program to perform the specified simulation experiment. Your program MUST:

(a) perform the experiment for (1) $n = 15,\ m = 15$, starting point (10,10), and (2) $n = 39,\ m = 19$, starting point (1,1);

(b) have an iteration limit, that is, a maximum number of squares that the bug may enter during the experiment. This ensures that your program will terminate. A maximum of 50,000 is appropriate for this exercise.

For each experiment, print (1) the total number of legal moves that the cockroach makes and (2) the final count array. This will show the "density" of the walk, that is, the number of times each tile on the floor was touched during the experiment.

General Information:

- Deadline : 2018/10/31 23:55.
- Upload your assignment to Moodle system.
- Upload file format : "student-ID_Name.rar" or "student-ID_Name.zip"
  Ex. "F12345678_王小明.rar"
- Your file should consist of the following items : Source Code & Readme file(Program description)
- Late homework will not be accepted.
- Any copies will be scored as zero. Do not plagiarize.

Example Output:

```
Experiment 1:
Total moves: 3010
  10   16   16   11    8   11   11    6   11    9   15   10   10   20    9
  13   26   17   13   17   15   12   13   16   16   19   17   14   21   14
   2   11   15   17   21   14    7   11   17   22   18   21   21   16   11
   3   18   19   23   21   15   14   14    8   16   17   16   16   19   15
  10   13   16   16   24   10   17   15   20   14    9   21   18   19   11
   9   18   18   22   18   23   13   10   11   16   13    5   11    9    9
  12   20   13   13   26   22   14    9   10   17   12   10    8    2    3
  16   13   16   11   15   20   17   16   11   11   16   10    2    8    3
   5   11   12   19   11    8   18   13   12   12   13   13    6    7    3
  11   14   15   17   18   13    9    7   11    8   11   10    3    2    4
  11   20   19   24   18   21   14    6   12   12    8    1    1    3    1
  12   14   26   13   22   24   16   12    8   11    5    4    3    2    3
  19   22   26   24   21   16   15   20   11    6    7    5   12    7    5
  19   41   33   21   21   24   24   17   20   16   12    7    2   10    6
  15   23   13   10   11   19   17   17   12   15    8    2    2    3    4
Experiment 2:
Total moves: 9037
   7    7   10    8    8    4    4    5    6    5    7   13   16   19   12    9    4    2    2
   8   14   13   11   12   10    6    5   10    5   18   15   14   18   21   11    8    7    2
  11   20   17   22   17   12   12   11    7   13   20   25   17   10   12   15   13    7    1
   7   21   20   24   20   14   15    9   11   22   30   31   15   10   10   11   16   12    5
  12   22   21   18   20   19    8   10   20   26   30   32   23   15   16   21   14   14    7
   9   21   15   16   18   10   15   18   19   24   30   22   15   14   21   11   13   15    7
  13    8   15   10   13   10   12    8    8   13   12   15   14   12   18   13    9    7    3
  10   17   12   10   17   15   11    6    8   10    8   15   10   10   17   18    8    5    1
   9    8    7   11   10   12    6    7    7    5    9   12   12   12   16    9    4   10    1
   5    8    5   10    8   12    5    7    9    4    7    7    6    8    8    9   11    4    5
   6    8    6   10   18   10   13    6    6    4    7    8    6    6    5    5    3    8    6
   6    6   10   12   18   18   12   10    5    5    7   10    8    7    7    4    4    3    2
   4    8    8   14   11    9    9    8   11    6    3    5    6    7    8    7    4    4    1
   4    6   12   11   10    5    5   11    9    3    7    3    7    6   11   10    8    3    1
   4   11   12    6   10    5    5    8    6    9    5    5   10   10   10   10   10    8    7
   9   12   11    9    6    4    7    7    8    6    4    8   14    9    9   11    9   13   15
   8   19   15    6    7    7    8    8   10    6    3    5    8   11   11   10    9   12    7
  18   17   17   10    4   11    7    9    9    8    5    3   11   10   13   16   15   14   11
  17   18   13   16   11   15   12    7   11   11    4    7    5   17   20   15   19   13    9
   8   16   16   12   15   14   10    9    9    9   13   10   16   13   11   18   19    7   10
   9   14    7   11   14   13   12    9   10   13    8    2    7    8    8   12   15   17   12
   9   12   19   12   10   12   14   10   14   14    9    7    4    6    9   12   17   17   13
  12   15   10   12   15   14   15   16   18   15    6   10    6    6   10   20   11   15   11
  13   18   16   19   19   15   13   18   10   12   10    4    6    8   12   14   23   15   10
  14   24   23   13   10    9   11   13    6    8   10    6    3    7    9   16   11   15    5
  20   31    8   19   16   10   11    9    5    5    9    8    7    7    8    6   13    8    8
  17   21   17   14   13   17   14    9    7    9    6    6   10   18   11    9    8   10   12
  10   17   19   16   13   13   11   11    9    9    5    5    9   12   13    8   10   10    9
  14   20   19    9   11   16    9    8   10    8    7    8   13   17    8    9    8   11    7
  12   25   17   10   13   15   19   13    9   15   11   16   16   17   20   17    6    7    5
  18   24   22   20   17   26   26   16   13   12   19   22   25   16   15   19    9    2    4
  12   27   21   21   21   25   12   14   14   18   20   29   19   23   15   11   12    9    2
  12   21   22   17   20   17   18   12   21   29   24   22   25   19   15   15   11   11    5
  19   21   29   18   13    8   15   17   16   20   21   15   20   15   11    8   14   14   12
  18   29   22   24   14   13   14   19   18   15    9   20   17   17   17   16   21   20    6
  24   29   21   24   18   16   15   20   11   16   11   12   11    7   11   14   24   22    7
  29   41   31   25   27   18   15   14   13   16   17   14    8   14   13   12   18   19   10
  19   37   29   28   20   15   13   17   14   18   17    9    9    9    6    9    9    9   11
  14   16   20   11    9   10   10   12    9    6    8    7    4    5    2    7    3    3    6
```

Note:

The content of the arrays will be different because k is randomly generated.

Please use formatted output for the arrays. Points will be deducted otherwise.