物件導向程式設計 第一次作業

Object-Oriented Programming Assignment 1

繳交截止日: 2019/4/2 23:59，如果在 2019/4/3-2019/10/9 之間繳交則算遲交，並酌扣 20%成績。這之後的時間均不接受補交。

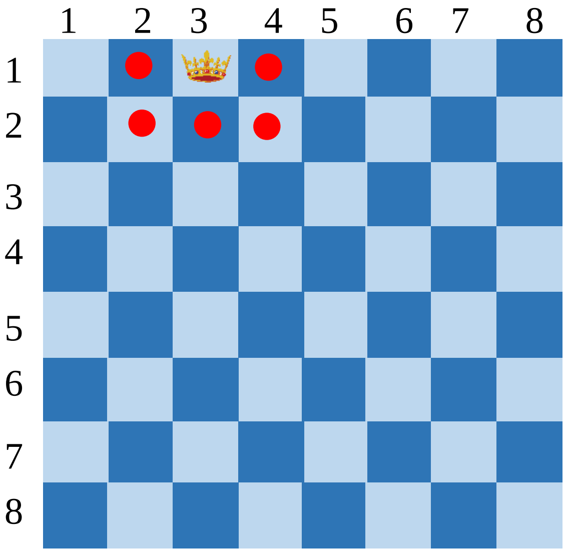
The submission deadline is 2019/4/2 23:59. You can have a late submission between 2019/4/3-2019/10/9 but can earn 80% credit. We do not accept even later submission.

1.

我們放置了皇后，在一個有 8 行 8 列的棋盤上。我們用(r_1, c_1)表示方格在 r_1 列 c_1 行。皇后在(r_1, c_1)上，若想移動到(r_2, c_2)，必須滿足$(r_1-r_2)^2+(c_1-c_2)^2 \le 2$ 才能移動。請計算當移動 *k* 次的時候，皇后可以到達多少格方格?
(行列的編號為 1 到 8)

A chessboard has 8 rows and 8 columns. We placed a Queen on a chessboard. Let's denote the cell in row r and column c by (r_1, c_1). A Queen on a cell (r_1, c_1) can move to another cell (r_2, c_2) if $(r_1-r_2)^2+(c_1-c_2)^2 \le 2$. Please count the number of cells that can be reached by the Queen in at most *k* moves.
(Row and column are numbered 1 through 8.)

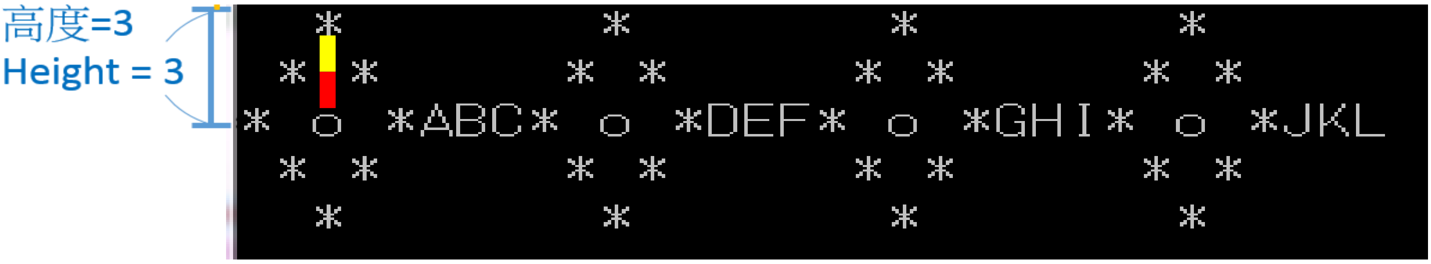
Input	**Output**
使用者輸入 *r, c, k* User can input *r, c, k*	
1 3 1	6 (包含原始位置) (including original position)



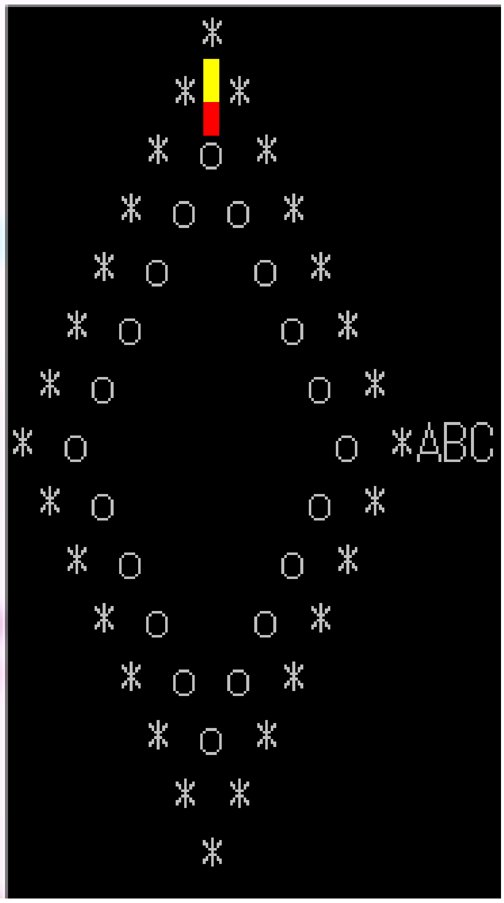
2.

給 2 個數字 *H* 與 *N*，請用*畫出 *N* 個高度為 H 的外圍菱形，距離外圍菱形 2 個單位長請用 o 畫出一個小菱形。每個外圍菱形後面都接三個遞增的英文字母，起始為 A，如下圖所示。

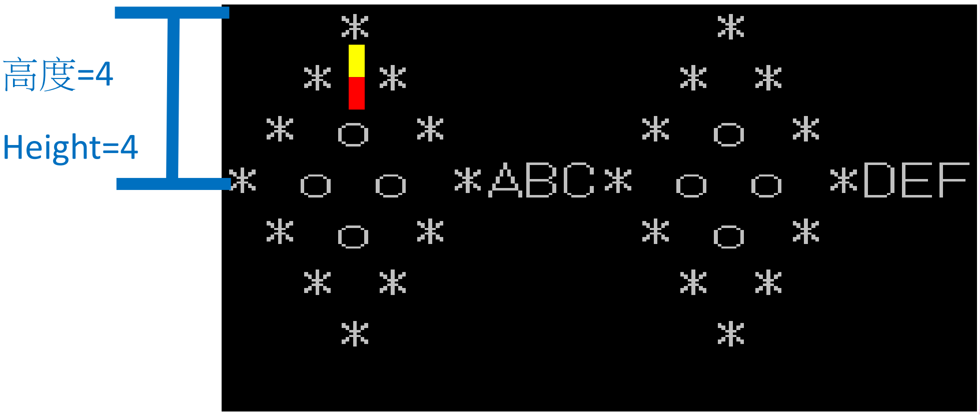
Given two numbers *H* and *N*, please use "*" to draw *N* diamonds with height *H*. Moreover, for each of *N* diamonds, please also use "o" to draw another smaller diamond with height *H-2* sharing the same center with the previous diamond. Each external diamond has three consecutive English alphabets in an increasing order, starting with A, as shown in below.



```
  *       *       *       *
 * *     * *     * *     * *
*  o  *ABC*  o  *DEF*  o  *GHI*  o  *JKL
 * *     * *     * *     * *
  *       *       *       *
```



```
       *
      * *
     * o *
    * o o *
   * o   o *
  * o     o *
 * o       o *
* o         o *ABC
 * o       o *
  * o     o *
   * o   o *
    * o o *
     * o *
      * *
       *
```



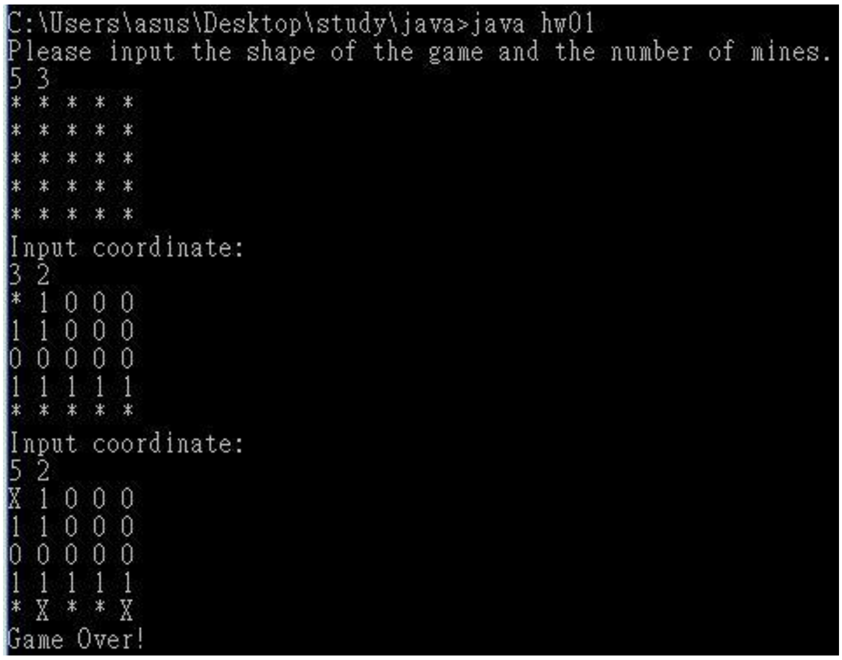
```
   *         *
  * *       * *
 * o *     * o *
* o o *ABC* o o *DEF
 * o *     * o *
  * *       * *
   *         *
```

3.

請實現一個簡單的踩地雷遊戲. 請輸入遊戲格數與地雷數量，其中地雷位置為隨機產生，此程式可以玩遊戲直到結束，如下圖所示。

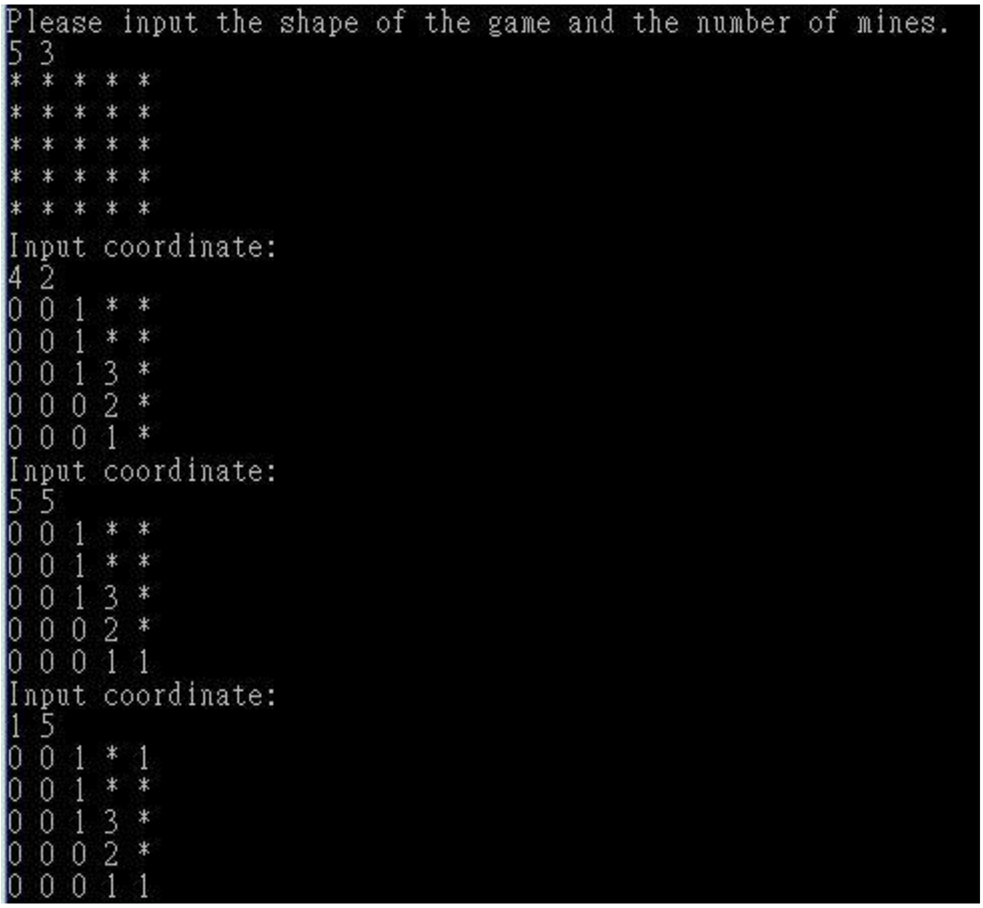
You are required to write a simple Minesweeper game. Please input the shape and size of the chessboard and the number of mines. The locations of the mines are randomly generated. Your program is required to be able to play the game until it's over. As shown in below.

(Example: Game Over)

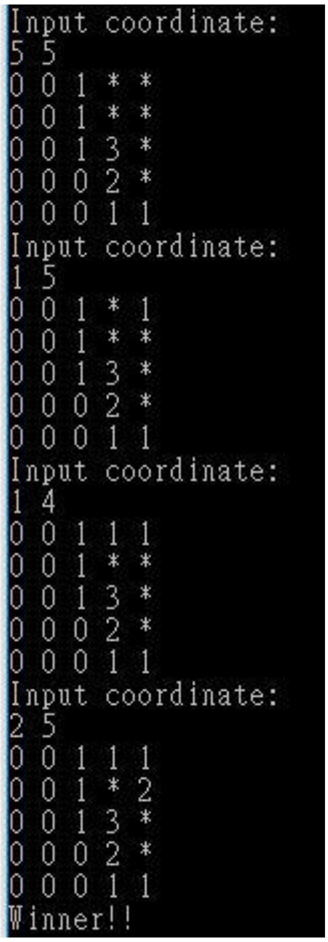


```
C:\Users\asus\Desktop\study\java>java hw01
Please input the shape of the game and the number of mines.
5 3
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
Input coordinate:
3 2
* 1 0 0 0
1 1 0 0 0
0 0 0 0 0
1 1 1 1 1
* * * * *
Input coordinate:
5 2
X 1 0 0 0
1 1 0 0 0
0 0 0 0 0
1 1 1 1 1
* X * * X
Game Over!
```

(Example: Win)



```
Please input the shape of the game and the number of mines.
5 3
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
Input coordinate:
4 2
0 0 1 * *
0 0 1 * *
0 0 1 3 *
0 0 0 2 *
0 0 0 1 *
Input coordinate:
5 5
0 0 1 * *
0 0 1 * *
0 0 1 3 *
0 0 0 2 *
0 0 0 1 1
Input coordinate:
1 5
0 0 1 * 1
0 0 1 * *
0 0 1 3 *
0 0 0 2 *
0 0 0 1 1
```



```
Input coordinate:
5 5
0 0 1 * *
0 0 1 * *
0 0 1 3 *
0 0 0 2 *
0 0 0 1 1
Input coordinate:
1 5
0 0 1 * 1
0 0 1 * *
0 0 1 3 *
0 0 0 2 *
0 0 0 1 1
Input coordinate:
1 4
0 0 1 1 1
0 0 1 * *
0 0 1 3 *
0 0 0 2 *
0 0 0 1 1
Input coordinate:
2 5
0 0 1 1 1
0 0 1 * 2
0 0 1 3 *
0 0 0 2 *
0 0 0 1 1
Winner!!
```