[2013 JAVA 物件導向程式設計 Homework 1]

注意事項

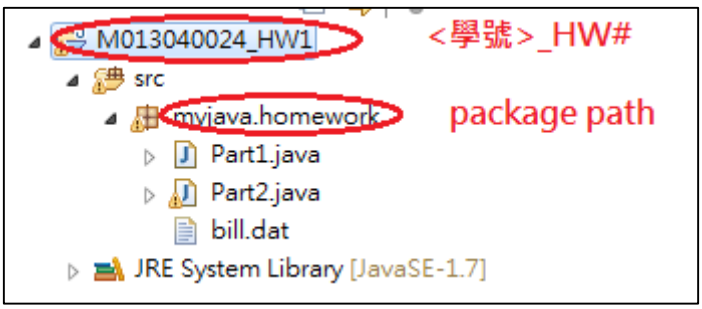
1. 請使用 JAVA 語言，配合 Eclipse 寫本次作業並進行測試，並安裝、使用 JAVA SE Development Kit(JDK) 7 函式庫。
2. 請依據作業規定設定 Eclipse 專案名稱與 package name，若未依照規定將根據狀況扣分。
3. 嚴禁抄襲其他同學作業，參與者(抄襲與被抄襲)均以零分計算。
4. 請對你的程式碼有深入瞭解，demo 時助教會問。
5. 對題目有問題可以寄信問助教群(java_ta@net.nsysu.edu.tw)或是到實驗室(EC5018)詢問，但不幫忙 debug。
6. 逾期以零分計算，不接受補交，有任何因素導致無法如期繳交，請事先告知；Demo 時間會另外通知。

作業規定與上傳

1. Eclipse 專案名稱：**<學號>_HW1**
2. Package path：**myjava.homework**
3. 作業請繳交專案之 tar 或 zip archive 並上傳至網路大學。

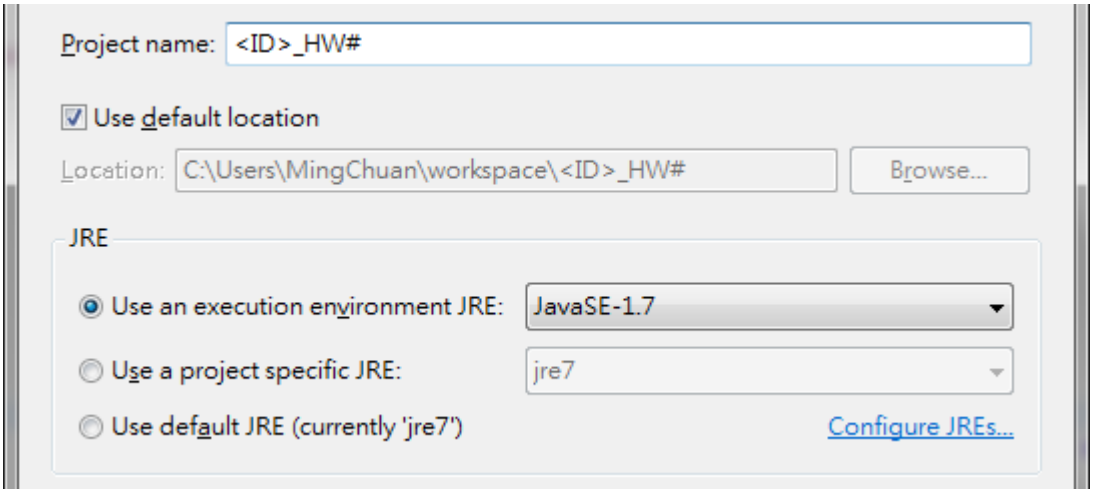
請於 **2014 年 3 月 3 日(週一) 23:59** 前上傳完畢，逾期以零分計算，不接受補交，有任何因素導致無法如期繳交，有問題請事先告知，再次強調，Demo 時間會另外通知。

4. Example of eclipse package explorer(請根據作業規定修改)：

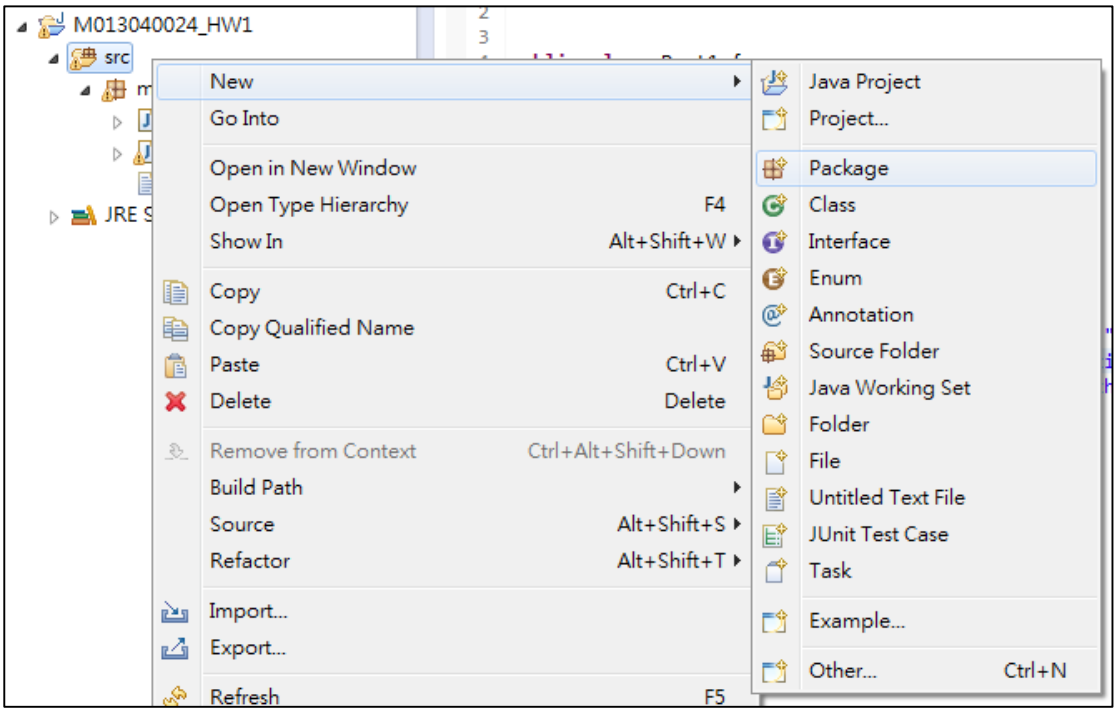


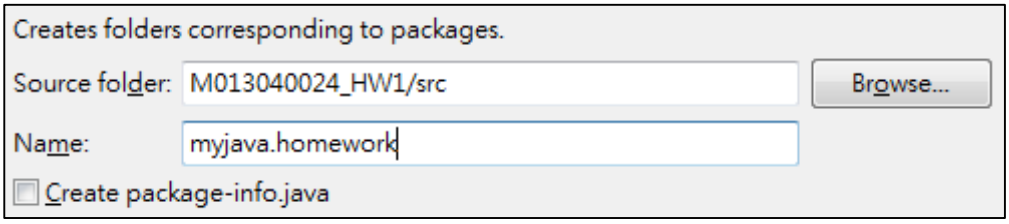
- **Eclipse 設定範例：**

1. 開新專案：File -> New -> Java Project，Project name 輸入你的學號與作業編號，底下 JRE 記得設定 JavaSE-1.7，Finish。

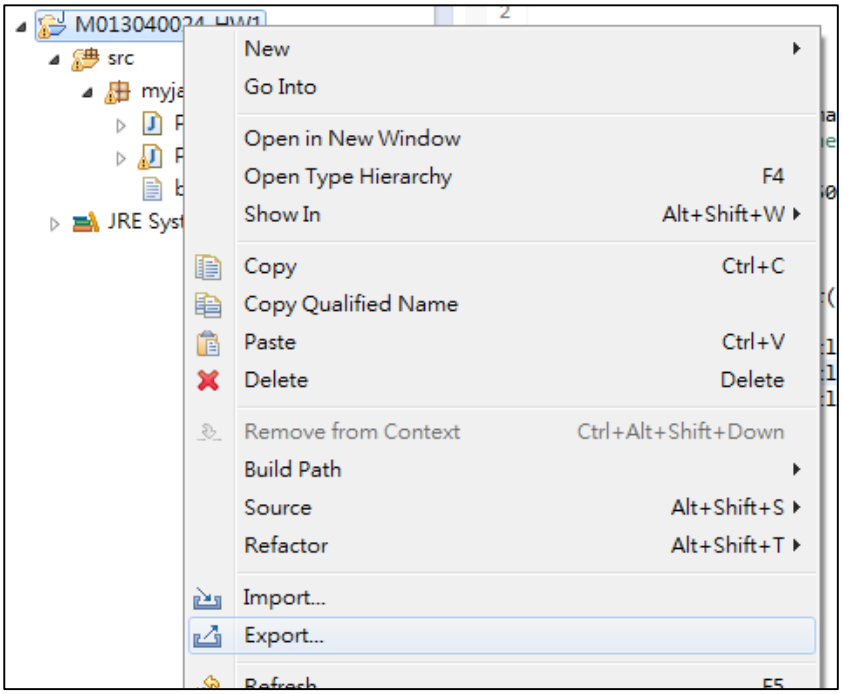


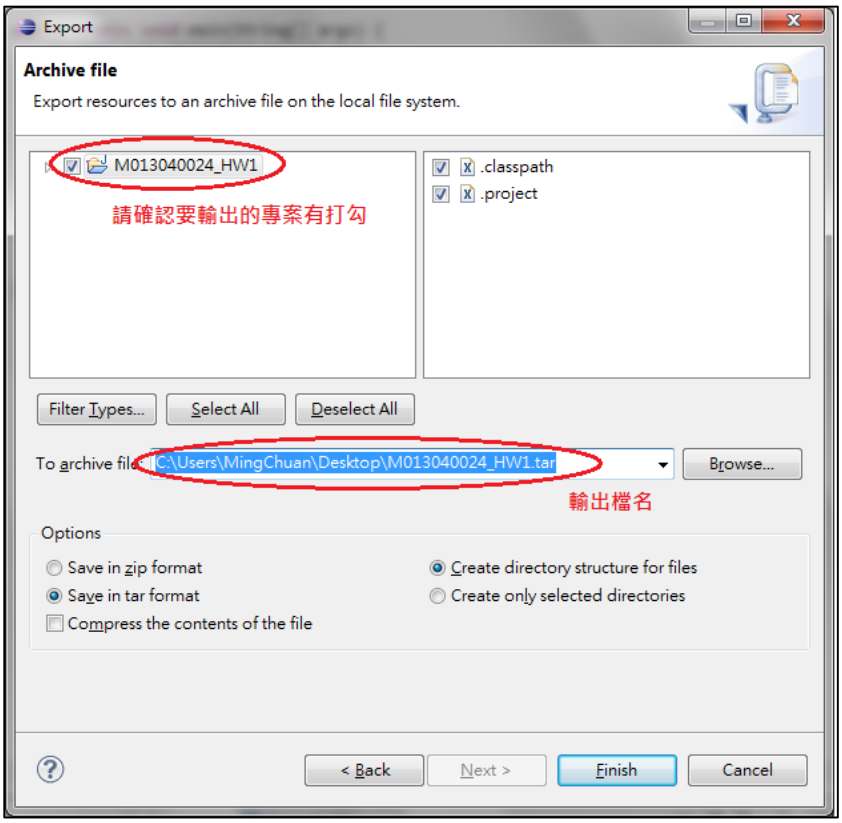
2. 建立 package：在專案中的 src 資料夾點右鍵 -> New -> Package，並把你本次作業的所有 code 與 data 全都放置在該 package 中。





3. 專案輸出：在專案上按右鍵 -> Export，選擇 General -> Archive File，輸出檔名請與專案名稱相同，格式 tar 或 zip 皆可。作業上傳至網路大學時，就上傳該 tar 或 zip 檔即可。





- **Part1**

The students of No.1 High School of Yibin Sichuan have a tradition: they throw books down the building to celebrate that end of semester at June. Now one of students throws a book like a free fall(自由落體) from 50 meters high. Please calculate how long will the book fall to the ground?

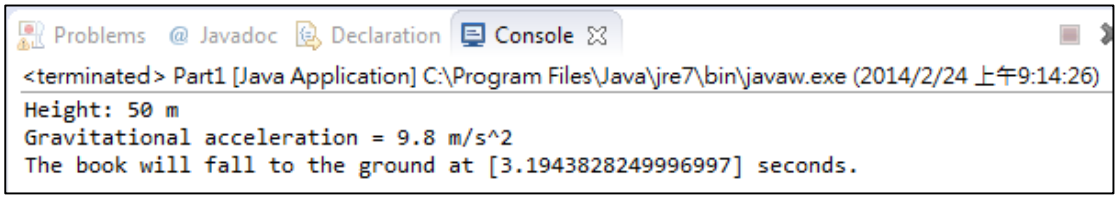
$$\Delta x = v_{avg}\Delta t = v_t\Delta t + g(\Delta t)^2 / 2$$

$$g = 9.8\ m/s^2$$

$$v_0 = 0\ m/s$$

You can use **Math.sqrt();** to square root of **Δ*t***.

Example:



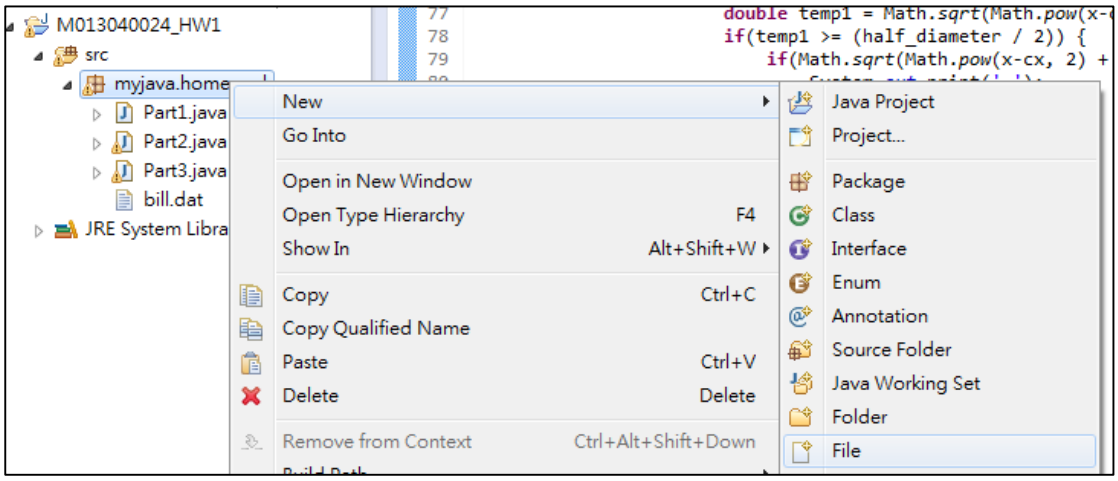
- **Part2**

Write a program that let user inputs the customer's name, member id, then reads the product's name, quantity, and price from file, separated by a semicolon ";". The name may contain spaces. Your job is output a bill with a tax rate of 12.5%. All prices should be output to two decimal places. The bill should be formatted in columns with 30 characters for the name, 15 characters for the quantity, 15 characters for the price, and 15 characters for the total.

First, ask user to input customer's name and member id:

```
Please input customer's name:
Jack (user key in)
Please input member id:
M023999999 (user key in)
```

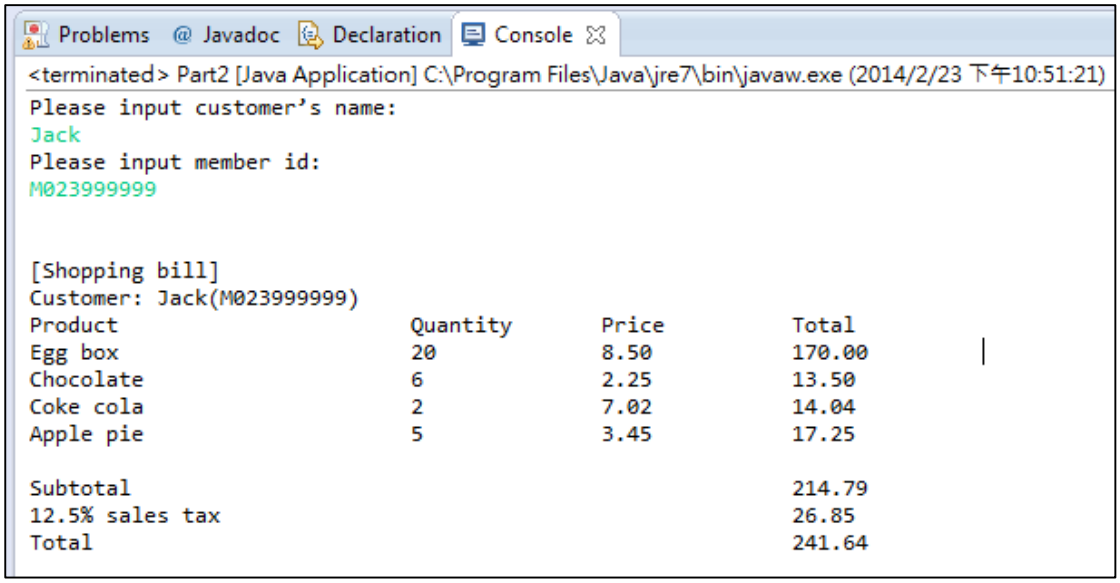
Then, read records from file. You should create a file named "bill.dat" in the same package and write down this content to your package. Your "bill.dat" can be read by path "src/myjava/homework/bill.dat".




```
Egg box;20;8.5
Chocolate;6;2.25
Coke cola;2;7.02
Apple pie;5;3.45
```

You can use **Integer.valueOf(String)** to convert String to Integer(object of int type), and **Double.valueOf(String)** to convert String to Double(object of double type).

Example:



- **Part3**

Please write a program that draws three of graphs, circle, donut, and two triangles. **You must use IF-ELSE, Switch, For loop or while to draw graphs.**

First, ask user to input the odd value, if user input even value, print error message and re-ask the question.

```
Please input the length of edge:
19 (user key in)
```

Second, ask user what graph he wants.

```
What image do you want?
    1) Circle
    2) Donut
    3) Two triangle
    4) Exit
```

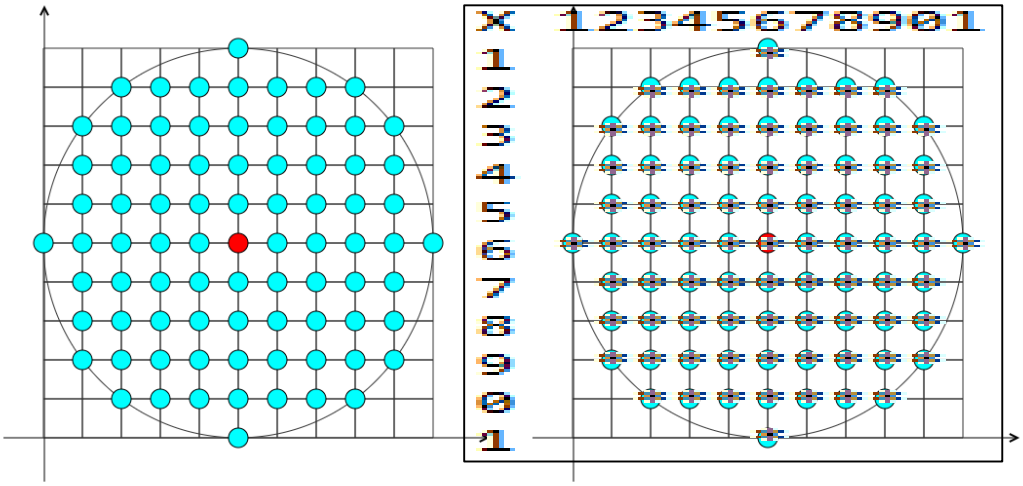
Then, output the graph.

There are some tips:

1. Circle: **Half-diameter = (length of edge - 1) / 2**

   You can calculate the distance from each point (X, Y) to center point.

   If distance <= half-diameter, print out “*”. Give an example, there is a circle which edge is 11 long. So Half-diameter is 5.



2. Donut: **inside circle’s half-diameter = (outside circle’s half-diameter) / 2**. Inside circle’s half-diameter may be **integer or float**. It will change the graph’s output. You choose one of them to match.

Example 1: (length = even numbers)

<Input deny>

```
Please input the length of edge:
2
Odd value only.
Please input the length of edge:
4
Odd value only.
Please input the length of edge:
6
Odd value only.
Please input the length of edge:
8
Odd value only.
Please input the length of edge:
10
Odd value only.
```

Example 2: (length = 5)

<Input access>

```
Please input the length of edge:
5
What graph do you want?
    1) Circle
    2) Donut
```

<Draw graph>

```
What graph do you want?
    1) Circle
    2) Donut
    3) Two traingle
    4) Exit
1
X 12345
1   *
2  ***
3 *****
4  ***
5   *
```

```
What graph do you want?
    1) Circle
    2) Donut
    3) Two traingle
    4) Exit
2
X 12345
1 ** **
2 * * *
3  ***
4 * * *
5 ** **
```

```
What graph do you want?
    1) Circle
    2) Donut
    3) Two traingle
    4) Exit
3
X 12345
1 *   *
2 ** **
3 *****
4 ** **
5 *   *
```

<Exit program>

```
What graph do you want?
    1) Circle
    2) Donut
    3) Two traingle
    4) Exit
4
Bye.
```

Example 3: (length = 7)

```
X 1234567
1    *
2  *****
3  *****
4 *******
5  *****
6  *****
7    *
```

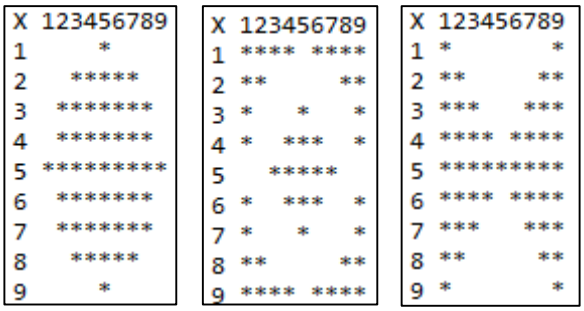
```
X 1234567
1 *** ***
2 *     *
3 *  *  *
4   ***
5 *  *  *
6 *     *
7 *** ***
```

OR

```
X 1234567
1 *** ***
2 *     *
3 * *** *
4   ***
5 * *** *
6 *     *
7 *** ***
```

```
X 1234567
1 *     *
2 **   **
3 *** ***
4 *******
5 *** ***
6 **   **
7 *     *
```

Example 3: (length = 9)

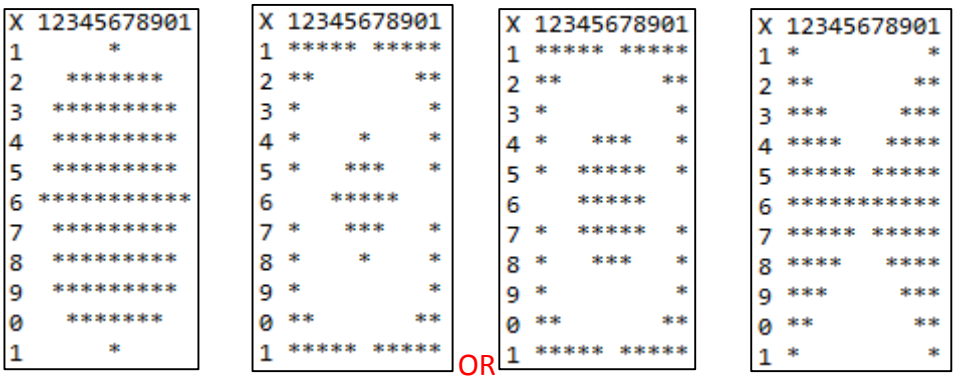


```
X 123456789
1     *
2   *****
3  *******
4  *******
5 *********
6  *******
7  *******
8   *****
9     *
```

```
X 123456789
1 ****  ****
2 **      **
3 *   *   *
4 *  ***  *
5   *****
6 *  ***  *
7 *   *   *
8 **      **
9 ****  ****
```

```
X 123456789
1 *       *
2 **     **
3 ***   ***
4 **** ****
5 *********
6 **** ****
7 ***   ***
8 **     **
9 *       *
```

Example 4: (length = 11)



```
X 12345678901
1      *
2   *******
3  *********
4  *********
5  *********
6 ***********
7  *********
8  *********
9  *********
0   *******
1      *
```

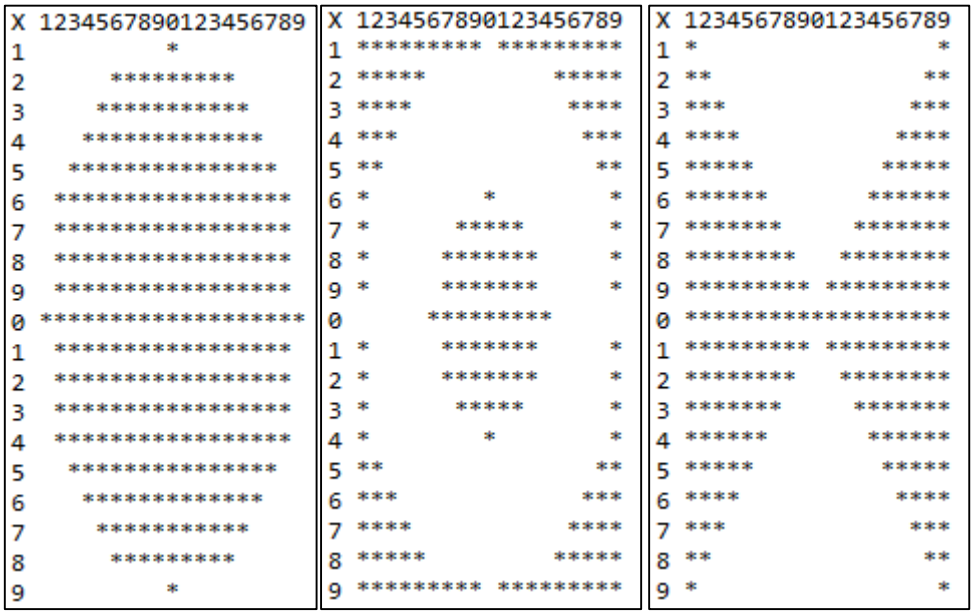
```
X 12345678901
1 *****  *****
2 **        **
3 *          *
4 *    *     *
5 *   ***    *
6    *****
7 *   ***    *
8 *    *     *
9 *          *
0 **        **
1 *****  *****
```

OR

```
X 12345678901
1 *****  *****
2 **        **
3 *          *
4 *   ***    *
5 *  *****   *
6    *****
7 *  *****   *
8 *   ***    *
9 *          *
0 **        **
1 *****  *****
```

```
X 12345678901
1 *         *
2 **       **
3 ***     ***
4 ****   ****
5 ***** *****
6 **********
7 ***** *****
8 ****   ****
9 ***     ***
0 **       **
1 *         *
```

Example 5: (length = 19)



```
X 1234567890123456789
1         *
2       *********
3      ***********
4     *************
5    ***************
6   *****************
7   *****************
8   *****************
9   *****************
0 *******************
1   *****************
2   *****************
3   *****************
4   *****************
5    ***************
6     *************
7      ***********
8       *********
9         *
```

```
X 1234567890123456789
1 *********  *********
2 *****          *****
3 ****            ****
4 ***              ***
5 **                **
6 *        *        *
7 *      *****      *
8 *     *******     *
9 *     *******     *
0       *********
1 *     *******     *
2 *     *******     *
3 *      *****      *
4 *        *        *
5 **                **
6 ***              ***
7 ****            ****
8 *****          *****
9 *********  *********
```

```
X 1234567890123456789
1 *                 *
2 **               **
3 ***             ***
4 ****           ****
5 *****         *****
6 ******       ******
7 *******     *******
8 ********   ********
9 ********* *********
0 *******************
1 ********* *********
2 ********   ********
3 *******     *******
4 ******       ******
5 *****         *****
6 ****           ****
7 ***             ***
8 **               **
9 *                 *
```