ชื่อ-นามสกุล นายภัทรภูมิ ตรีพงษ์ชัย รหัสประจำตัวนักศึกษา 6604062636151 *ปีการศึกษา 2567*

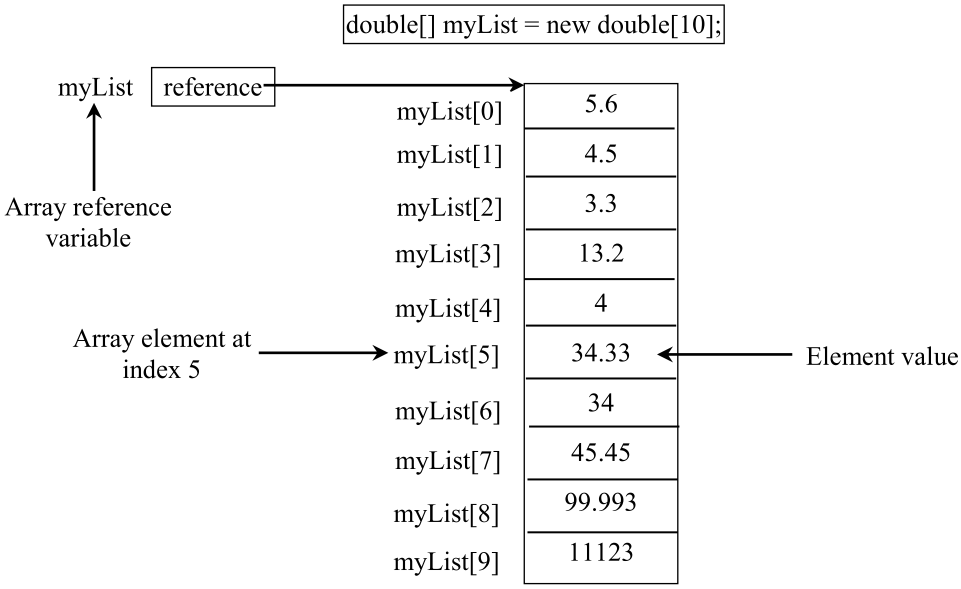
แบบฝึกหัดปฏิบัติการคาบที่ 4: Array

คำสั่ง

1. ให้ศึกษาการสร้างอาร์เรย์จากตัวอย่างต่อไปนี้

อาร์เรย์คือโครงสร้างข้อมูลที่มีชนิดข้อมูลประเภทเดียวกัน

การประกาศอาร์เรย์ของชนิดข้อมูลพื้นฐาน มีรูปแบบดังนี้



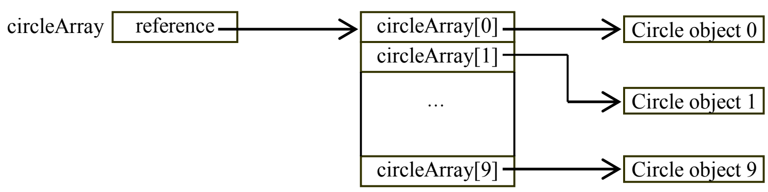
อาร์เรย์ของวัตถุ (Array of Objects)

- ตัวอย่างของการประกาศอาร์เรย์ของวัตถุ

    Circle[] circleArray = new Circle[10];

- อาร์เรย์ของวัตถุหรือเรียกอีกอย่างว่าอาร์เรย์ของตัวแปรอ้างอิงวัตถุ(*array of reference variables*)
- การเรียกใช้ circleArray[1].findArea() จะทำงานใน 2 ขั้นตอนคือ
    - การประกาศ circleArray จะเป็นการอ้างอิงไปยังอาร์เรย์ทั้งหมด
    - ส่วน circleArray[1] จะอ้างอิงไปยังวัตถุของคลาส Circle
- ตัวอย่างเมื่อประกาศ Circle[] circleArray = new Circle[10]; จะมีรูปแบบของการอ้างอิงดังนี้

    Circle[] circleArray = new Circle[10];



ชื่อ-นามสกุล นายณัฐโชติ ทรัพย์ชัย รหัสประจำตัวนักศึกษา 6604062636151 *ปีการศึกษา 2567*

---

1. ให้ศึกษาและทดลองพิมพ์ Code ตัวอย่างการสร้างอาร์เรย์จากตัวอย่างต่อไปนี้

```
public class MinMaxDemo{
    public static void main(String[] args){
        int a[] ={-128, 65, -235, 99, 0, 26};
        int minIndex= findMinIdx(a);
        //int maxIndex= findMinIdx(a);
        System.out.println("min value is a["+minIndex+"]="+a[minIndex]);
        //System.out.println("max value is a"+maxIndex+"]="+a[maxIndex]);

    }
    public static int findMinIdx(int[] a){
        int k, minIdx=0;
        for(k=1;k<a.length;k++){
            if(a[k]<a[minIdx])
            {
                minIdx=k;
            }
        }
        return minIdx;

    }
}
```

{-128, 65, -235, 99, 0, 26}

1.1 ผลลัพธ์ของโปรแกรมคือ

min value is a[2]= -235

1.2 ให้อธิบายการทำงานของ public `static int findMinIdx(int[] a)`

โปรแกรมจะทำการหาค่าที่น้อยที่สุดที่อยู่ Index ไหน และจะ return Index นั้นกลับไป

1.3 ให้เพิ่มการทำงานของเมธอด `public static int findMaxIdx(int[] a)` สำหรับหาตำแหน่ง index ของอาร์เรย์ที่มีค่ามากที่สุด

```
public static in findMaxIdx(int[] a){
    int k, maxIdx=0;
    for(k=1; k<a.length; k++{
        if(a[k] > a[maxIdx])
        {
            maxIdx =k;
        }
    }
    return maxIdx;
}
```

ชื่อ-นามสกุล นายณัฐภัทร์ โนนด ตรีมงคลชัย รหัสประจำตัวนักศึกษา 6604062636151 *ปีการศึกษา 2567*

2. จงอธิบายว่าเหตุใดโปรแกรมด้านล่างจึง compiles ไม่ผ่าน

โปรแกรม	ผลลัพธ์ของโปรแกรม

```
public class Test {
   public static void main(String[] args) {
      double[100] r;

      for (int i = 0; i < r.length(); i++);
         r(i) = Math.random * 100;
   }
 }
```

1. ไม่มีการใช้ [100] มีแค่ [ ] และก็ขาดการ new double[100]
2. มีการปิดลูปแล้วตามจริงต้องใช้ { เพื่อครอบ Code ที่ต้องการให้ทำงาน
3. r(i) ไม่มี แค่ r[i]
4. ต้องเป็น Math.random()
5. ต้องเป็น r.length

```
public class Test {
   public static void main(String[] args) {
      int list[] = {1, 2, 3, 4, 5, 6};
      for (int i = 1; i < list.length; i++)
         list[i] = list[i - 1];

      for (int i = 0; i < list.length; i++)
         System.out.print(list[i] + " ");
   }
 }
```

output: 1 1 1 1 1 1

```
public class Test {
   public static void main(String[] args) {
      int number = 0;
      int[] numbers = new int[1];
      m(number, numbers);
      System.out.println("number is " + number
      + " and numbers[0] is " + numbers[0]);
   }
public static void m(int x, int[] y) {
   x = 3;
   y[0] = 3;
   }
}
```

output: number is 0 and numbers[0] is 3

```
public class Test {
   public static void main(String[] args) {
      int[] list = {1, 2, 3, 4, 5};
      reverse(list);
      for (int i = 0; i < list.length; i++)
         System.out.print(list[i] + " ");
   }
   public static void reverse(int[] list) {
      int[] newList = new int[list.length];
      for (int i = 0; i < list.length; i++)
         newList[i] = list[list.length - 1 - i];
         list = newList;
      }
}
```

output: 1 2 3 4 5

```
public class Test {
 public static void main(String[] args) {
   int[][] array = {{1, 2}, {3, 4}, {5, 6}};
   for (int i = array.length - 1; i >= 0; i—) {
      for (int j = array[i].length - 1; j >= 0; j—)
         System.out.print(array[i][j] + " ");
      System.out.println();
   }


  }
}
```

output: 6 5
4 3
2 1

```
public class Test {
   public static void main(String[] args) {
      int[][] array = {{1, 2, 3, 4}, {5, 6, 7, 8}};
      System.out.println(m1(array)[0]);
```

ชื่อ-นามสกุล นายณัฐโชน ศรีพงษ์ชัย รหัสประจำตัวนักศึกษา 6604062636157 *ปีการศึกษา 2567*

```
    System.out.println(m1(array)[1]);
  }
  public static int[] m1(int[][] m) {
     int[] result = new int[2];
    result[0] = m.length;
     result[1] = m[0].length;
     return result;
  }
}
```

output: 2
4

3.จากส่วนของ code ต่อไปนี้

```
class Test2{
    public static void main(String[] args){
       int[] list= {1,9,3,7,2};
        list=dosomething(list);
     }
    public static int[] dosomething(int[] input){
        int temp;
        for (int i = 1; i < input.length; i++) {
            for(int j = i ; j > 0 ; j--){
                if(input[j] < input[j-1]){
                    temp = input[j];
                    input[j] = input[j-1];
                    input[j-1] = temp;
                }
            }
        }
        for(int i=0;i<input.length;i++){
          System.out.print(input[i]+" ");
        }
        return input;
    }
}
```

จากโปรแกรมต่อไปนี้ให้แสดงผลลัพธ์ของโปรแกรมในแต่ละรอบของการทำงานของ Loop i พร้อมแสดงผลลัพธ์สุดท้าย

ค่าในอาร์เรย์ `list` เริ่มต้น 1, 9, 3, 7, 2

ภายในเมธอด `public static int[] dosomething(int[]input)`

i= 1

ค่าในอาร์เรย์ input

1	9	3	7	2

i= 2

ค่าในอาร์เรย์ input

1	3	9	7	2

i= 3

ค่าในอาร์เรย์ input

1	3	7	9	2

i= 4

ค่าในอาร์เรย์ input

1	2	3	7	9

i=

ค่าในอาร์เรย์ input

ผลลัพธ์สุดท้ายของโปรแกรม

1, 2, 3, 7, 9

4 กำหนดให้ A[0...n-1] เป็นชุดข้อมูลที่เป็นจำนวนเต็ม n จำนวน จงเขียนโปรแกรมแบบ OOP ในการเรียงข้อมูล A[0...n-1] จากน้อยไปมาก พร้อมทั้งหาความถี่สะสมของข้อมูลแต่ละตัว ตัวอย่างเช่น A= [9 5 9 5 8]

ข้อมูลที่เรียงจากน้อยไปมาก	5	8	9
ความถี่ของข้อมูลแต่ละตัว	2	3	5

ให้นักศึกษานิยาม Class ชื่อ AscendSortFreq ที่ประกอบไปด้วย

- Data fields ชนิด double[ ] A ซึ่งแสดงชุดข้อมูล จำนวน n ตัว
- Constructor ที่กำหนดค่าให้แก่ double[ ] A จำนวน n ตัว
- Method AscendSort(double[ ] A) ที่ return array B ที่เรียงจากน้อยไปมาก
- Method SortCommuFreq(double[ ] B) ที่ return array C ความถี่ของข้อมูลแต่ละตัว

5. [MatrixMultiplication] ให้เขียน class MatrixMultiplication ซึ่งทำการคำนวณหาผลคูณของสองเมทริกซ์ การคูณเมทริกซ์ทำได้เมื่อจำนวนคอลัมภ์ของเมทริกซ์แรกมีค่าเท่ากับจำนวนแถวของเมทริกซ์ที่สอง ดังนั้นถ้าเมทริกซ์ A เป็นเมทริกซ์ขนาด NxL และเมทริกซ์ B มีขนาดเป็น LxM แล้วผลลัพธ์ของการคูณเมทริกซ์คือ C=AxB จะมีขนาดเป็น NxM โดยที่สมาชิกแต่ละตัวของเมทริกซ์ C มีค่าดังสมการต่อไปนี้

$$c_{ik} = a_{i1}b_{1k} + a_{i2}b_{2k} + a_{i3}b_{3k} + .. + a_{iL}b_{Lk}$$

ตัวอย่าง ให้ $A = \begin{bmatrix} 1 & 2 \\ -1 & 0 \\ 3 & 2 \end{bmatrix}, B = \begin{bmatrix} 1 & 5 & 2 \\ -2 & 0 & 1 \end{bmatrix}$

วิธีทำ $AB = \begin{bmatrix} 1 & 2 \\ -1 & 0 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} 1 & 5 & 2 \\ -2 & 0 & 1 \end{bmatrix}$

ใช้หลักการแถว คูณ หลัก

$$= \begin{bmatrix} (1)(1)+(2)(-2) & (1)(5)+(2)(0) & (1)(2)+(2)(1) \\ (-1)(1)+(0)(-2) & (-1)(5)+(0)(0) & (-1)(2)+(0)(1) \\ (3)(1)+(2)(-2) & (3)(5)+(2)(0) & (3)(2)+(2)(1) \end{bmatrix}$$

$$= \begin{bmatrix} -3 & 5 & 4 \\ -1 & -5 & -2 \\ -1 & 15 & 8 \end{bmatrix}$$

ข้อมูลนำเข้า ขนาดของเมทริกซ์ A และข้อมูลในเมทริกซ์ A
ขนาดของเมทริกซ์ B และข้อมูลในเมทริกซ์ B

ข้อมูลส่งออก ผลคูณของสองเมทริกซ์

ข้อมูลนำเข้า	ข้อมูลส่งออก
3 2 1 2 -1 0	-3 5 4 -1 -5 -2 -1 15 8

3 2 2 3 1 5 2 -2 0 1	

6. จงเขียนโปรแกรมเพื่อคำนวณหาคู่จุดที่ใกล้เคียงกันมากที่สุด

ข้อมูลนำเข้า บรรทัดแรกรับจำนวนจุด n จุด

n บรรทัดถัดไปแสดงพิกัดของจุดแต่ละจุด

ข้อมูลส่งออก ระยะทางคู่จุดที่ใกล้เคียงกันมากที่สุด

ข้อมูลนำเข้า	ข้อมูลส่งออก
8 3 3 3 1- 1 1 2 4 0.5 2 1- 1- 0.5- 4 2 -1	1.12

7. minTwoSet

กำหนดให้ จำนวนเต็ม n จำนวน โดยมีอยู่ m จำนวนอยู่ในกลุ่ม A และ n-m คนอยู่ในกลุ่ม B จงเขียนโปรแกรมหาคำตอบของผลต่างของผลรวมในแต่ละกลุ่มน้อยที่สุดสามารถหาได้จากสมการ

$$\min\left(\left|\sum_{i=1, i\in A}^{m} w_i - \sum_{j=1, j\in B}^{n-m} w_j\right|\right)$$

$w_i \in n, w_j \in n$ โดย และ $|\ \ |$ คือ ค่าสัมบูรณ์ เช่น $|-3| = 3$ หรือ $|3| = 3$

เขียนโปรแกรมหาค่าผลต่างของผลรวมในแต่ละกลุ่มน้อยที่สุด (เขียนแบบ OOP)

ข้อมูลนำเข้า อ่านจาก Standard Input

บรรทัดแรก ระบุ จำนวนn โดย $3 \le n < 1000$

บรรทัดสอง ระบุค่า w_i จำนวนเต็มบวก n จำนวนที่ไม่ซ้ำกัน โดยแต่ละจำนวนคั่นด้วยช่องว่าง

ข้อมูลส่งออก ส่งออกไปยัง Standard Output

แสดงผลลัพธ์ของผลต่างของผลรวมในแต่ละกลุ่มน้อยที่สุด

ตัวอย่างข้อมูล

ข้อมูลนำเข้า	ข้อมูลส่งออก
4 3 101 99 1	0
5 14 16 47 25 2	6
6 7 2 1 3 6 0	1

8. [**Find Pokemon**] ญาญ่าผู้เล่นหน้าใหม่ของเกมส์โปเกมอนต้องการจับตัวปิกาจู เนื่องจากปิกาจูเป็นตัวที่หาได้ยากมาก ดังนั้นญาญ่าจึงไปรวบรวมสถิติการเกิดตัวโปเกมอน ณ พิกัดต่าง ๆ โดยพบว่าในแต่ละพิกัดจะมีความถี่ของการเกิดตัวโปเกมอนตัวต่าง ๆ ที่ไม่เท่ากัน โดยมีการเก็บข้อมูลเป็นรูปขนาด HxW ช่อง โดยญาญ่าต้องการหาปิกาจูจากรูปนี้

ตัวอย่างของรูปขนาด 4x5 แสดงเป็นตารางด้านล่าง กำหนดตารางชื่อ A ตัวเลขในแต่ละช่องแสดงความถี่ของการเกิดตัวโปเกมอนในช่องนั้น

5	1	2	10	4
4	30	3	0	100
3	25	10	4	10
3	20	4	8	5

ในการหาตำแหน่งของปิกาจูจะมีเงื่อนไข 3 ข้อดังนี้

1. ปิกาจูจะปรากฏเป็น 2 ช่องติดกันพอดี
2. สองช่องที่เป็นบริเวณที่มีปิกาจูควรมีค่าความถี่ของการเกิดตัวโปเกมอนในช่องนั้น ต่างกันไม่เกิน 10
3. ตำแหน่งของปิกาจูน่าจะเป็นตำแหน่งที่มีความถี่ของการเกิดตัวโปเกมอนสูงคือต้องเป็นสองช่องที่มีผลรวมของค่าความถี่ของการเกิดตัวโปเกมอนมากที่สุด

จากตารางตำแหน่งที่ตรงตามเงื่อนไขคือ **A[2][2] และ A[3][2]** จงเขียนโปรแกรมที่รับตารางแสดงความถี่ของการเกิดตัวโปเกมอน จากนั้นให้หาตำแหน่งมุมบนซ้ายของช่องที่น่าจะเกิดปิกาจูมากที่สุด โดยระบุแถวและคอลัมภ์ช่องนั้น

ข้อมูลนำเข้า

บรรทัดแรก ระบุขนาดตาราง HxW

บรรทัดที่ 2 ถึง H+1 แสดงความถี่ของการเกิดตัวโปเกมอนในแถวที่ i โดยระบุเป็นจำนวนเต็มจำนวน W ตัว

จำนวนที่ j จะเป็นความถี่ของการเกิดตัวโปเกมอนในช่องที่อยู่ในคอลัมภ์ j

ข้อมูลส่งออก

มีบรรทัดเดียว คือ มุมบนซ้ายของช่องที่น่าจะเกิดตัวปิกาจูมากที่สุดโดยระบุแถวและคอลัมภ์ช่องนั้น

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
4 5 5 1 2 10 4 4 30 3 0 100 3 25 10 4 10 3 20 4 8 5	2 2
4 4 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0	3 2

9. จงเขียนเมธอด isConsecutiveFour(**int**[][] values) เพื่อตรวจสอบอาร์เรย์สองมิติต่อไปนี้มีตัวเลขตัวเดียวกันเรียงต่อกันครบสี่ตัวหรือไม่ ในแนวตั้ง แนวนอน หรือแนวทะแยง

public static boolean isConsecutiveFour(**int**[][] values)

จากนั้นให้เขียนโปรแกรมทดสอบที่ผู้ใช้สามารถป้อนจำนวนแถวและคอลัมภ์ของอาร์เรย์สองมิติและป้อนค่าภายในอาร์เรย์ โดยเมื่อป้อนค่าไปแล้วให้เรียกใช้เมธอด isConsecutiveFour(**int**[][] values) โดยการคืนค่า true กรณีที่ภายในอาร์เรย์มีตัวเลข ตัวเรียงต่อกัน กรณีอื่นให้คืน 4ค่าเป็น false

ตัวอย่าง เมื่อส่งอาร์เรย์ต่อไปนี้เข้าไปที่เมธอดจะคืนค่า true ออกมา

0	1	0	3	1	6	1
0	1	6	8	6	0	1
5	6	2	1	8	2	9
6	5	6	1	1	9	1
1	3	6	1	4	0	7
3	**3**	**3**	**3**	4	0	7

0	1	0	3	1	6	1
0	1	6	8	6	0	1
5	**5**	2	1	8	2	9
6	**5**	6	1	1	9	1
1	**5**	6	1	4	0	7
3	**5**	3	3	4	0	7

0	1	0	3	1	6	1
0	1	6	8	6	0	1
5	6	2	1	**6**	2	9
6	5	6	**6**	1	9	1
1	3	**6**	1	4	0	7
3	**6**	3	3	4	0	7

0	1	0	3	1	6	1
0	1	6	8	6	0	1
9	6	2	1	8	2	9
6	**9**	6	1	1	9	1
1	3	**9**	1	4	0	7
3	3	3	**9**	4	0	7

ข้อมูลนำเข้า จำนวนแถวและคอลัมภ์ของอาร์เรย์สองมิติและป้อนค่าภายในอาร์เรย์

ข้อมูลส่งออก คืนค่า 1 กรณีที่ภายในอาร์เรย์มีตัวเลข ตัวเรียงต่อกัน กรณีอื่นให้คืนค่าเป็น 0 4

ข้อมูลนำเข้า	ข้อมูลส่งออก
7 6 0 1 0 3 1 6 1 0 1 6 8 6 0 1 5 6 2 1 8 2 9 6 5 6 1 1 9 1 1 3 6 1 4 0 7 3 3 3 3 4 0 7	1

ชื่อ-นามสกุล....นาย[illegible] [illegible]....รหัสประจำตัวนักศึกษา....6604062636151.... *ปีการศึกษา 2567*

10. (Car) ขับรถหลบสิ่งกีดขวาง

ในการแสดงขับรถผาดโผนบนถนนที่มีเลนทั้งหมด m เลน โดยให้หมายเลขประจำเลนจากซ้ายไปขวามีค่าตั้งแต่ 1 จนถึง m ตามลำดับ นักแสดงขับรถผาดโผนต้องบังคับรถให้แล่นไปบนถนนดังกล่าวให้ปลอดภัยตลอดระยะเวลา t หน่วย การแสดงเริ่มต้น ณ เวลา 0 ถึง t นักแสดงขับรถผาดโผนอยู่ในเลนที่ n ในแต่ละ 1 หน่วยเวลา อาจมีสิ่งกีดขวางตกลงมายังถนนบางเลน ทำให้เขาต้องบังคับรถเพื่อหลีกเลี่ยงสิ่งกีดขวาง ซึ่งมีทางเลือกในการบังคับรถอยู่ 3 แบบ ได้แก่ 1 หมายถึง การเปลี่ยนเลนไปทางซ้าย 1 เลนในเวลาถัดไปไปยังเลนที่มีหมายเลขประจำเลนน้อยกว่า 2 หมายถึงการเปลี่ยนเลนไปทางขวา 1 เลนในเวลาถัดไป (ไปยังเลนที่มีหมายเลขประจำเลนมากกว่า) และ 3 หมายถึง การขับอยู่ในเลนเดิม กำหนดให้ถนนเป็นเส้นตรงตลอดทาง จงเขียนโปรแกรมเพื่อบังคับให้รถแล่นไปตามเส้นทางนี้โดยปลอดภัย โดยชุดข้อมูลทดสอบจะมีคำตอบที่ถูกต้องเพียง 1 คำตอบเสมอ

ข้อมูลนำเข้า

1. บรรทัดแรกระบุจำนวนเลน m โดยที่ $2 \le m \le 40$
2. บรรทัดที่สองระบุหมายเลขเลนเริ่มต้น $1 \le n \le m$
3. บรรทัดที่สามระบุระยะเวลา t โดยที่ $1 \le t \le 100$
4. บรรทัดที่สี่ถึงบรรทัดที่ t+3 แสดงสถานะของถนน ณ เวลา t=1, 2, ,K ตามลำดับ แต่ละบรรทัดระบุตัวเลข m ตัวเลขแต่ละตัวแสดงสถานะของถนนตั้งแต่เลนที่ 1 ถึงเลนที่ mโดยเลข 0 หมายถึงเลนนั้นไม่มีสิ่งกีดขวาง และเลข 1 หมายถึงมีสิ่งกีดขวางอยู่

ข้อมูลส่งออก

มีอยู่ t บรรทัด แต่ละบรรทัดมีตัวเลข 1 ตัวเพื่อแสดงถึงทางเลือกในการบังคับรถของนักแสดงขับรถผาดโผนในแต่ละช่วงเวลา บรรทัดที่ i หมายถึงการเปลี่ยนเลนจากเวลาที่ i-1 ไปยังเวลาที่ i เมื่อ i=1, 2, , .. t โดยที่เลข 1 จะหมายถึงขับไปทางซ้าย 1 เลน, เลข 2 หมายถึงขับไปทางขวา 1 เลน, และเลข 3 หมายถึงขับอยู่ในเลนเดิม

ตัวอย่างที่ 1

ข้อมูลนำเข้า	ข้อมูลส่งออก
7 5 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 1 0 1 1 1 1 1	1 1 1 1 2

ชื่อ-นามสกุล.....นายณัฐ[illegible] ทรัพย์[illegible].....รหัสประจำตัวนักศึกษา.....6604062636151..... *ปีการศึกษา 2567*

---

11. **[Oil Deposits] บ่อน้ำมัน**

นักธรณีวิทยาต้องการสำรวจหาแหล่งน้ำมันบนพื้นที่สี่เหลี่ยมขนาดใหญ่เรียกว่า GRID โดยบริษัทได้ตีเส้นใน GRID ให้อยู่ในรูปของตาราง และทำการสำรวจโดยการใช้เครื่องมือสำหรับตรวจจับน้ำมันว่าพื้นที่สี่เหลี่ยมที่สำรวจมีน้ำมันอยู่หรือไม่ บริเวณที่มีน้ำมันจะเรียกว่า pocket ถ้ามี pocket ในตารางแต่ละช่องต่อกันไม่ว่าจะเป็นแนวตั้ง แนวนอน แนวทะแยง จะถือว่าเป็นบ่อน้ำมันบ่อเดียวกัน งานที่ต้องทำคือให้ตอบคำถามว่าในพื้นที่ พื้นที่ขนาดใหญ่ 1GRID มีบ่อน้ำมันกี่บ่อ

ข้อมูลนำเข้า

ข้อมูลนำเข้าประกอบด้วย หรือมากกว่า 1 1GRID แต่ละ GRID จะเริ่มต้นด้วยตัวเลข m และ n ซึ่งแทนด้วยจำนวนแถวและจำนวนคอลัมภ์ใน GRID กรณีที่ m=0 แสดงว่าไม่มีการรับข้อมูลต่อ กำหนดให้ $1 \le m \le 100$ และ $1 \le n \le 100$ แถวต่อไปจำนวน m แถวจะมีแถวละ n ตัวอักษร แต่ละตัวอักษรแทนด้วยสถานะของน้ำมัน ณ ตารางดังกล่าวโดย `*' แทนไม่มีน้ำมัน `@' แทน oil pocket.

ข้อมูลส่งออก

สำหรับแต่ละ GRID จะแสดงจำนวนของบ่อน้ำมันที่ค้นพบ

ข้อมูลนำเข้า	ข้อมูลส่งออก
1 1 *	0
3 5 *@*@ **@** *@*@*	1
1 8 @@****@*	2
5 5 ****@ *@@*@ *@**@ @@@*@ @@**@	2