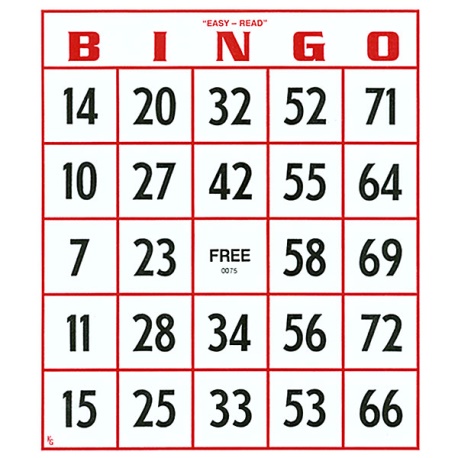
**Bingo**

**Overview:**

The standard Bingo card is a 5 by 5 table containing integers between 1 and 75, with a free spot at the center that contains 0.



The first column in the table contains only integers between 1 and 15, the second column numbers are all between 16 and 30, the third are 31 to 45, the fourth 46-60, and the fifth 61-75. There are no duplicate numbers on a Bingo card.

There are about 155 × 145 × 135 × 125 × 11 4 possible arrangements of the numbers on a bingo card.

Before the game starts, each player is given a Bingo card. The game also has a "caller", who issues a list of all the integers between 1 and 75 in random order. As each number is called, players mark the appropriate squares on their cards. The middle square, the free square, is always marked.

The winner of the game is the first person whose Bingo card has one of the following winning conditions:

* All five spots in a column are marked
* All five spots in a row are marked
* All the spots in either of the two diagonals are marked
* The four corner squares are all marked

Read the Wikipedia page on [Bingo](http://en.wikipedia.org/wiki/Bingo_(US)).

**Objectives:**

* Reinforce the concept of object-oriented programming;
* Gain experience of working with two-dimensional arrays.
* Gain experience with reading and writing text files

**Instructions:**

**Part I:**

Implement a method

public void write(String outputFile)

that writes a random Bingo card configuration and a stream of random numbers (which you will generate by the shuffle() method below) between 1 and 75 to the output file. The first five lines of the output represent the Bingo card. Each line is a row of five integers separated by spaces. The third number in the third line, the free square, is always contain the number 0. Next will come seventy five integers, all in one line, in random order. For example,

3 18 34 60 75

12 26 38 47 73

2 23 0 49 66

15 21 43 58 67

6 22 31 56 62

59 69 7 39 32 44 27 53 21 25 28 9 17 61 12 37 5 55 [et cetera, until all numbers have been listed]

The first column in the Bingo card configuration must contain only integers between 1 and 15; the second column numbers are all between 16 and 30, the third are 31 to 45, the fourth 46-60, and the fifth 61-75. There are no duplicate numbers on a Bingo card.

You are also to implement

public void shuffle(ArrayList list)

that shuffles a given list of numbers. You shall use this method to generate a list of random integers within the write() method. Note, you cannot call the Collections.shuffle() method.

**Part II:**

Implement a method

public void read(String inputFile)

that reads an input file created by the above method write(). You will store a Bingo card configuration (the first five lines) in a two-dimensional array of integers, called card. A list of 75 integers should be stored in the array stream.

Implement a method

public int playGame()

that returns the first integer from the stream array that gives you the earliest possible winning condition, i.e, the last number called before you would declare Bingo. The winning conditions are defined in the Overview.

We provide two sample bingo inputs for testing purposes: [test1.txt](http://www.andrew.cmu.edu/course/15-121/labs/HW-2%20Bingo/test1.txt) should return 55, and [test2.txt](http://www.andrew.cmu.edu/course/15-121/labs/HW-2%20Bingo/test2.txt) should return 69.

### Starter Code:

Download The following files

* [MainDriver.java](http://www.andrew.cmu.edu/course/15-121/labs/HW-2%20Bingo/MainDriver.java)
* [Bingo.java](http://www.andrew.cmu.edu/course/15-121/labs/HW-2%20Bingo/Bingo.java)
* [input.txt](http://www.andrew.cmu.edu/course/15-121/labs/HW-2%20Bingo/input.txt)