

**ITCS 6114 Fall 2016**  
**Programming Homework 1**  
**Points: 100**

**This assignment is to be completed individually. No group work is allowed.**

**Assignment Description**

Implement an algorithm to find all local peaks in a  $m \times n$  matrix.

Your program structure should be similar to the peak finding problem set that was introduced in the course.

The main program takes an input file (plain text file) of any  $m \times n$  matrix and outputs the result to a text file name "localPeaks\_Output.txt"

Your program should compile and run from the command line. For example, if you implement your code in java, it should run using following command assuming running in the default folder- the location of your source files:

```
javac FindLocalPeaks.java
java findLocalPeaks localPeaks_Input.txt
```

The input file "localPeaks\_Input.txt" should be a plain text file and it contains a matrix representation using an array of arrays representation where each element (array) represents a row in that matrix corresponding to its index.

For example, consider this 10 x 10 matrix

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

The content of the corresponding input file would be:

```
problemMatrix = [[1, 2, 3, 4, 5, 6, 7, 8, 9, 10], [11, 12, 13, 14, 15, 16, 17, 18, 19, 20],
[21, 22, 23, 24, 25, 26, 27, 28, 29, 30], [31, 32, 33, 34, 35, 36, 37, 38, 39, 40],
[41, 42, 43, 44, 45, 46, 47, 48, 49, 50], [51, 52, 53, 54, 55, 56, 57, 58, 59, 60],
[61, 62, 63, 64, 65, 66, 67, 68, 69, 70], [71, 72, 73, 7, 75, 76, 77, 78, 79, 80],
[81, 82, 83, 84, 85, 86, 87, 88, 89, 90], [91, 92, 93, 94, 95, 96, 97 ,98, 99, 100]]
```

Note that the name of this input file can be different so DO NOT hard code this in your code. It has to be read from the command line! When grading, different input files will be used.

The output of your program should be a plain text file called "localPeaks\_Output.txt" that includes all the row and column references for all local peaks in the corresponding input matrix. The output file should be saved in the same folder as the program's default folder. Programming Languages and IDE

## **Grading Criteria**

The total of 100 points for this project is broken up into:

- 20 points for proper construction of data structures required in the program.
- 20 points for correctly handling the input and output files.
- 40 points for efficiently implementing the local peak finding algorithm.
- 20 points for compilation, structure, and documentation (a readme file and comment in the code).

Within these criteria, your grade will be based on program structure, efficiency, and correct execution. The structure of your code will be judged for quality of the comments, quality of the data structure design, and especially the logic of the implementation. The comments need not to be extremely long: just explain clearly the purpose of each class and each function within each class.

## **Submission Guidelines**

Your submission must include all your source code files and a brief report as a README file. DO NOT include any IDE-specific project files, any compiled files, or any executable files. Every file should have your name in a comment line at the top. Your README file should have a brief description of your program design, the breakdown of the algorithm, the compiler you used, the platform you used, a summary of what you think works and fails in your program, and a short description of your data structure design.

You will submit your project on Canvas. You should submit a single zip file containing your source code files and README file. You can submit your project multiple times; only the most recent project submission will be graded. No late submission is allowed for this project and no submission through email will be considered.