**CS311 DSA Project 1**

**Your Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Univ. ID: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Instructions**

* Demo deadline: lab class today, Submission deadline: check **due date** on course webpage
* Remember to comment your code
* Remember to take screenshots of the running results
* Zero mark will be given if your program does not compile, or gets into an infinite loop (does not terminate)
* Submission guide
  + Create a new folder named with CS311\_Proj1\_yourID
  + Archive your code files and this .doc file into **CS311\_Proj1\_yourID.zip (or .rar)**
  + Submit **the archived file** to cs\_scu@foxmail.com

**Problems (Peak Finding)**

**Definition**: (One-dimensional Peak)

Let be an array of integers of length , integer is a ***peak*** if adjacent integers are not large than .

*Example*: is a peak.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
| 4 | 3 | 9 | 10 | **14** | 8 | 7 | 2 | 5 | 6 |

**Task**: Write algorithm with the following properties:

* Input: An integer array of length
* Output: Any position such that is a peak (might be more than one peak)

***Remarks***: 1. You need to consider at least THREE different , that is, at least three test examples.

2. The used data structures MUST include at least one of the list, stack, or queue.

**PROJECT REPORT**

**Inputs** (including screenshots)

Command: Input first, then input a set of integers with the length .

**Outputs** (including screenshots)