**SpecialStack**

**Q:** Design a Data Structure SpecialStack that supports all the stack operations like push(), pop(), isEmpty(), isFull() and an additional operation getMin() which should return minimum element from the SpecialStack. To implement SpecialStack, you should only use standard Stack data structure and no other data structure like arrays, list, .. etc.

**Test Cases:**

**Input:** 5 1 8 4 9

**Output:** 1

**Input:** 52 87 14 32 9

**Output:** 9

**HINT:** Maintain two stacks: actual stack and an auxilliary stack.

* Actual stack will store all the actual values pushed in the stack
* Auxilliary stack will store the minimum value of the values stored in the actual stack.

**Algorithm:**

* When push( ) is called

--> **if** the new element is greater than the top of the auxilliary stack than push the element which is already present at the top of the auxilliary stack in the auxilliary stack and new element in the actual stack.

--> **else** push the new element to the auxilliary stack and also to the actual stack.

* When pop( ) is called the element present at the top of both actual and auxiliary stacks are removed.
* When peek( ) is called the element at the top of the auxilliary stack is returned.
* When min( ) is called the element present at the top of the auxilliary stack is returned.

**Example:**

Input: 5 1 8 4 9

* 5 will be pushed to both actual and auxilliary stcaks.
* Push 1 to both the stacks.
* Push 1 to auxilliary and 8 to the actual stack.
* Push 1 to the auxilliary stack and 4 to the actual stack.
* Push 1 to the auxilliary and 9 to the actual stack.
* When min() is called, 1 is returned.