

Mohammad Ali Jinnah University

Chartered by Government of Sindh - Recognized by HEC

**Lab Task 6**

**Name:** Muhamad Fahad

**Id:** FA19-BSSE-0014

**Subject:** Data Structures and Algorithms Lab (CS 2511)

**Lab Title:** Stack

**Section:** AM

**Teacher:** MUHAMMAD MUBASHIR KHAN

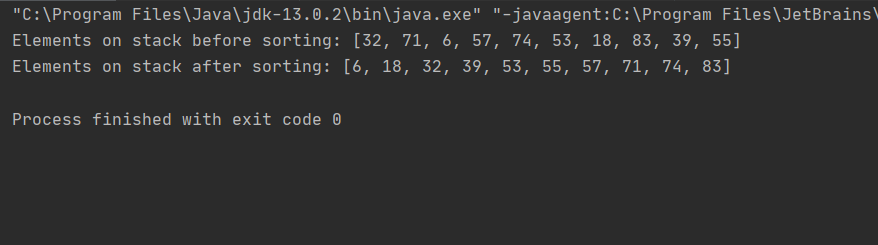
**Date:** Tuesday, December 1, 2020

**Q1) Sort an stack by using another stack.**

**Code:**

class Question01{  
  
 public static void main(String args[]) {  
 Random rand = new Random();  
 Stack input = new Stack();  
 for (int i = 0; i < 10; i++) {  
 input.push(rand.nextInt(100));  
 }  
  
  
 System.*out*.println("Elements on stack before sorting: "+ input.toString());  
  
 input = *sorting*(input);  
  
 System.*out*.println("Elements on stack after sorting: "+ input.toString());  
  
 }  
  
  
 public static Stack<Integer> sorting(Stack<Integer> stack) {  
 int lenght = stack.size();  
 int tempVar, min ;  
 Stack<Integer> Final = new Stack<Integer>();  
 Stack<Integer> temp = new Stack<Integer>();  
  
  
 for (int i = 0; i < lenght; lenght--) {  
 min = stack.peek();  
 for (int j = 0; j < lenght; j++) {  
   
 if(stack.isEmpty()){  
 j = lenght;  
 continue;  
 }  
  
 if (min > stack.peek())  
 min = stack.peek();  
  
 temp.push(stack.pop());  
 }  
  
 for (int j = 0; j < lenght; j++)  
 if (min != (tempVar = temp.pop()))  
 stack.push(tempVar);  
 else  
 Final.push(tempVar);  
 }  
  
 return Final;  
 }  
}

**Output:**

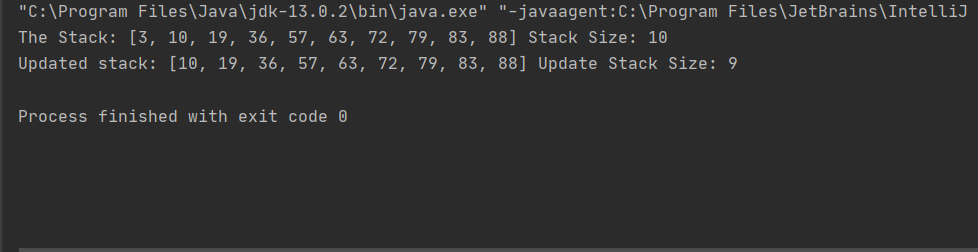
****

**Q2) Delete a specific element from an stack.**

**Code:**

class Question02{  
 public static void main(String[] args) {  
 Random rand = new Random();  
 Stack input = new Stack();  
  
 for (int i = 0; i < 5; i++) {  
 input.push(rand.nextInt(100));  
 }  
  
 input = Question01.*sorting*(input);  
  
 System.*out*.println("The Stack: "+input+" Stack Size: "+input.size());  
 input = *deleteSmallest*(input);  
 System.*out*.println("Updated stack: "+input+" Update Stack Size: "+input.size());  
  
 }  
  
 public static Stack deleteSmallest(Stack<Integer> stack) {  
 int lenght = stack.size();  
 int tempVar, min;  
  
 Stack<Integer> temp = new Stack<Integer>();  
  
 min = stack.peek();  
 for (int j = 0; j < lenght; j++) {  
 if (!(stack.isEmpty()) && min > stack.peek())  
 min = stack.peek();  
  
 temp.push(stack.pop());  
 }  
  
 for (int j = 0; j < lenght; j++)  
 if (min != (tempVar = temp.pop()))  
 stack.push(tempVar);  
  
 return stack;  
 }  
}

**Output:**

****