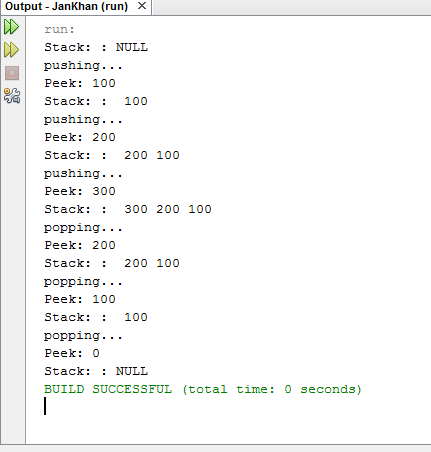
Arifullah Jan - 186943

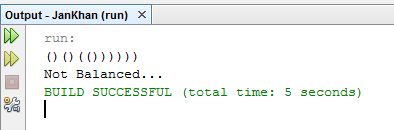
BSCS 6A

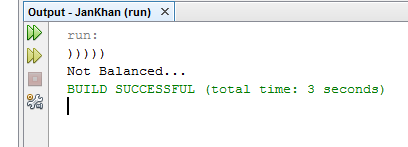


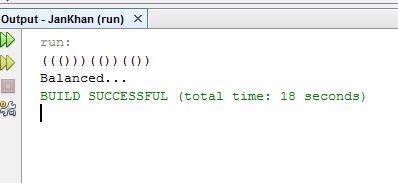
/\*  
 \* To change this license header, choose License Headers in Project Properties.  
 \* To change this template file, choose Tools | Templates  
 \* and open the template in the editor.  
 \*/  
  
/\*\*  
 \*  
 \* @author arifu  
 \*/  
public class JanKhan {  
 static Stack myStack=new Stack(5);  
   
 public static void main(String[] args){  
 System.out.println("Stack: : "+myStack);  
 myStack.push(100);  
 System.out.println("pushing...");  
 System.out.println("Peek: "+myStack.peek());  
 System.out.println("Stack: : "+myStack);  
 myStack.push(200);  
 System.out.println("pushing...");  
 System.out.println("Peek: "+myStack.peek());  
 System.out.println("Stack: : "+myStack);  
 myStack.push(300);  
 System.out.println("pushing...");  
 System.out.println("Peek: "+myStack.peek());  
 System.out.println("Stack: : "+myStack);  
 myStack.pop();  
 System.out.println("popping...");  
 System.out.println("Peek: "+myStack.peek());  
 System.out.println("Stack: : "+myStack);  
 myStack.pop();  
 System.out.println("popping...");  
 System.out.println("Peek: "+myStack.peek());  
 System.out.println("Stack: : "+myStack);  
 myStack.pop();  
 System.out.println("popping...");  
 System.out.println("Peek: "+myStack.peek());  
 System.out.println("Stack: : "+myStack);  
 }  
  
}

/\*  
 \* To change this license header, choose License Headers in Project Properties.  
 \* To change this template file, choose Tools | Templates  
 \* and open the template in the editor.  
 \*/  
  
/\*\*  
 \*  
 \* **@author** arifu  
 \*/  
public class Stack {  
 int[] arr;  
 int index=-1;  
 Stack (int size){  
 this.arr=new int[size];  
 }  
 public void push(int n){  
 index++;  
 arr[index]=n;  
 }  
 public int peek(){  
 if(index<0){  
 return 0;  
 }  
 return arr[index];  
   
 }  
 public boolean pop(){  
 if(index<0){  
 return false;  
 }  
 arr[index]=0;  
 index--;  
 return true;  
 }  
  
 @Override  
 public String toString() {  
 String temp="";  
 if(index==-1){  
 temp+="NULL";  
 }  
 else{  
 for(int i=index; i>=0;i--){  
 temp+=" "+arr[i];  
 }  
   
 }  
 return temp;  
 }  
  
}

## Task 1b:







public class JanKhan {  
 static Stack myStack=new Stack(5);  
   
 public static void main(String[] args){  
 Scanner in=new Scanner(System.in);  
 String parths= in.next();  
   
 if(tester(parths)){  
 System.out.println("Balanced...");  
 }  
 else{  
 System.out.println("Not Balanced...");  
 }  
   
 }  
 public static boolean tester(String str){  
 Stack stack=new Stack(str.length());  
 for(int i=0;i<str.length();i++){  
 if(str.charAt(i)=='('){  
 stack.push(1);  
 }  
 else if(str.charAt(i)==')'){  
 if(!stack.pop()){  
 return false;  
 }  
   
 }  
   
 }  
 return true;  
 }  
}

# TASK 2

# 

public class JanKhan {  
 static Stack myStack=new Stack(5);  
   
 public static void main(String[] args){  
 Scanner in=new Scanner(System.in);  
 String parths= in.next();  
   
 if(tester(parths)){  
 System.out.println("palindrome");  
 }  
 else{  
 System.out.println("Not a palindrome...");  
 }  
   
 }  
 public static boolean tester(String str){  
 Stack stack=new Stack(str.length()\*2);  
 Queue queue = new Queue(str.length()\*2);  
 for(int i=0; i<str.length();i++){  
 int temp=(int)str.charAt(i);  
 stack.push(temp);  
 queue.enQueue(temp);  
   
   
 }  
 for(int i=0;i<str.length();i++){  
 if(stack.peek()!=queue.front()){  
 return false;  
 }  
 stack.pop();  
 queue.deQueue();  
 }  
 return true;  
 }  
   
  
}

# Queue.java

public class Queue {  
   
   
 int back=-1;  
 int front=-1;  
 private int SIZE;  
 private int[] queue;  
  
 public Queue(int SIZE) {  
 this.SIZE = SIZE;  
 this.queue=new int[SIZE];  
   
 }  
   
  
 public void enQueue(int value){  
 if(back == SIZE-1)  
 back=-1;  
 else{  
 if(front == -1){  
 front = 0;  
 }  
   
 back++;  
 queue[back] = value;  
   
 }  
 }  
 public void deQueue(){  
 if(front == back){  
 //System.out.println("\nQueue is Empty!!! Deletion is not possible!!!");  
 }   
 else{  
 front++;  
   
 }  
 }  
   
 public int front(){  
 return queue[front];  
 }  
 public int back(){  
 return queue[back];  
 }  
   
  
}