NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL

OOPS LAB ASSIGNMENT-3

Name: P MANOHAR RAO

Roll No:197158

Section: A

1) Write a Java Program to Implement Binary Search Algorithm

```
import java.util.*;
class search{
int binarysearch(int []a,int l,int r,int num)
{
if(l>r)
{
return -1;
}
int i=(l+r)/2;
if(a[i]==num)
{
return i;
}
else if(a[i]>num)
{
return binarysearch(a,l,i-1,num);
}
else
{
return binarysearch(a,i+1,r,num);
}
}
}
public class a3q1 {
public static void main(String[] args) {
System.out.println("enter no of elements");
Scanner sc=new Scanner(System.in);
int k=sc.nextInt();
int[] a=new int[k];
```

```
System.out.println("enter nos");
for(int i=0;i<k;i++){
a[i]=sc.nextInt();
}
System.out.println("enter reqiured number");
int num=sc.nextInt();
search s=new search();
System.out.println(s.binarysearch(a,0,k-1,num));
}
}
```

```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac a3q1.java } ; if ($?) { java a3q1
enter no of elements
5
enter nos
1
6
5
9
3
enter reqiured number
2
-1
```

2) Write a Java Program to Compute all the permutations of the string.

```
import java.util.*;
class permute{
void print(String s,String ans)
{ if(s.length()==0){
System.out.println(ans+" "); return;
}
for(int i=0;i<s.length();i++){
char c=s.charAt(i);
String r=s.substring(0,i)+s.substring(i+1);
print(r,c+ans);
}
}
}
public class a3q2 {
public static void main(String[] args) {
Scanner sc=new Scanner(System.in);
String s=sc.nextLine();
permute p=new permute();
p.print(s,"");
}
```

```
}
```

```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac a3q2.java } ; if ($?) { java a3q2 }
CSE
ESC
SEC
ECS
CES
SCE
CSE
```

3) Write a Java Program to Implement multiple inheritance

```
interface I1{
default void show(){ System.out.println("Printing I1");
}
}

interface I2 {
default void show(){ System.out.println("Printing I2");
}
}

class testclass implements I1,I2 {
public void show()
{
I1.super.show();
I2.super.show();
}

public static void main(String args[])
{
testclass d = new testclass(); d.show();
}
}
```

```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac testclass.java } ; if ($?) { java testclass }
Printing I1
Printing I2
PS C:\Users\91961\Documents\CODES>
```

4) Write a Java Program to Detect loop in a LinkedList

```
import java.util.*;
class ll{
int data;
ll node;
```

```
ll head=null;
ll tail=null;
ll(int data){
this.data=data;
this.node=null;
}
void addlist(int data1){
if(head==null){
head=new ll(data1);
head=tail;
}
else{
ll temp=new ll(data1);
tail.node=temp;
tail=temp;
}
}
void link(ll head,int k){ ll temp=head;
for(int i=0;i<k;i++){
temp=temp.node;
}
this.tail.node=temp;
}
boolean verify(ll head){ ll t1=head,t2=head; while(true){
t1=t1.node; t2=t2.node.node; if(t1==t2){
return true;
}
}
}
}
public class a3q4 {
public static void main(String[] args) {
ll node1=new ll(0);
Scanner sc=new Scanner(System.in);
System.out.println("enter no of elements");
int k=sc.nextInt();
for(int i=0;i<k;i++){
int q=sc.nextInt();
node1.addlist(q);
}
System.out.println("enter the index");
int e=sc.nextInt();
```

```
node1.link(node1,e);
node1.verify(node1);
}
}
```

```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac a3q4.java } ; if ($?) { java a3q4 }
enter no of elements
5
1
2
3
4
5
enter the index
3
```

5) Write a Java Program to Implement Binary Tree Data Structure and count the leaves

```
import java.util.*;
class binarytree{
Scanner sc=new Scanner(System.in);
class node{
int data;
node left;
node right;
node(int data){
data=data;
left=right=null;
}
}
node root=null;
node add(node B,int k){
if(B==null){
B=new node(k);
}
else{
int v=sc.nextInt();
if(v!=0){
B.left=add(B.left,v);
}
v=sc.nextInt();
if(v!=0){
B.right=add(B.right,v);
}
}
return B;
```

```
}
void insert(int d){ root=add(root,d);
}
void in(node B){ if(B==null) return;
System.out.println(B.data+" ");
in(B.left);
in(B.right);
}
void inorder(){ in(root);
System.out.println();
}
int leaf(node b){ if(b==null){
return 0;
}
int count=0;
if(b.left==null && b.right==null){ count+=1;
}
count=count+leaf(b.left)+leaf(b.right);
return count;
}
}
public class a3q5{
public static void main(String args[]){ binarytree b=new binarytree
();
Scanner sc=new Scanner(System.in);
int k=sc.nextInt();
b.insert(k);
b.inorder();
sc.close();
}
}
```

6) Write a Java Program to Merge two lists

```
import java.util.ArrayList;
import java.util.Arrays;
public class a3q7{
public static void main(String[] args) throws Exception
{
```

```
ArrayList<String> listOne = new ArrayList<>(Arrays.asList("a1", "a
2", "a3", "a4"));

ArrayList<String> listTwo = new ArrayList<>(Arrays.asList("a5", "a
6", "a7", "a8"));
listOne.addAll(listTwo);
System.out.println(listOne);}}
```

```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac a3q7.java } ; if ($?) { java a3q7 }
[a1, a2, a3, a4, a5, a6, a7, a8]
```

7) Write a java program that implements correctly the producer consumer problem using multithreading

```
class Q{
int n;
boolean val=false; synchronized int get()
{
if (!val) {
try { wait();
} catch (InterruptedException e) { System.out.println(e);
}
}
System.out.println("got: "+n); val=false;
notify(); return n;
}
synchronized void put(int n)
{
if (val) {
try { wait();
} catch (InterruptedException e) { System.out.println(e);
}
}
val=true; notify(); this.n=n;
System.out.println("put:"+n);
}
}
class producer implements Runnable
{
Q q; Thread t;
producer(Q q)
```

```
{
this.q=q;
t=new Thread(this,"producer"); t.start();
}
public void run() { int i=1;
int k=10; while(k>0){
q.put(i++);
k--;
}
}
}
class consumer implements Runnable {
Q q;
Thread t;

consumer(Q q) { this.q = q;
t = new Thread(this, "consumer");
t.start();
}

public void run() { int k = 10; while (k > 0) {
q.get();
k--;
}
}

}
public class a3q8{
public static void main(String[] args) { Q q=new Q();
new producer(q); new consumer(q);
}
}
```

```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac a3q8.java } ; if ($?) { java a3q8 }
put:1
got: 1
put:2
got: 2
put:3
got: 3
put:4
got: 4
put:5
got: 5
put:6
got: 6
put:7
got: 7
put:8
got: 8
put:9
got: 9
put:10
got: 10
```

8) Write a java program that implements the multithreaded application that has four threads. First thread generates 'n' number of random integers (user defined), the time elapse betweeneach random generation should be 2 seconds, if the generated number is odd, then second thread computes the cube of that number and prints it, if the number generated by first threadis even then third thread computes the square of the number and prints it. Now, Fourth threadhas to sum up all the generated numbers and prints it

```
import java.util.*;
class mainthread implements Runnable
{
int n; Thread t;
mainthread(int n)
{
this.n=n;
t=new Thread(this,"mainthread"); t.start();
}
public void run()
{
int[] numbers=new int[n]; Random rand=new Random(); int i;
for(i=0;i<n;i++)
{
numbers[i]=rand.nextInt(100); if (numbers[i]%2==0) {
new displaysquare(numbers[i]);
}
else
{
```

```
new displaycubes(numbers[i]);
}
try { Thread.sleep(2000);
} catch (InterruptedException e) { System.out.println(e);
}
}
new displaysum(numbers);
}
}
class displaysquare implements Runnable
{
Thread t; int n;
displaysquare(int n)
{
this.n=n;
t=new Thread(this,"displaysquare"); t.start();
}
public void run(){ n=n*n;
System.out.println("square: "+n);
}
}
class displaycubes implements Runnable
{
Thread t; int n;
displaycubes(int n)
{
this.n=n;
t=new Thread(this,"displaycube"); t.start();
}
public void run()
{
n=n*n*n; System.out.println("cube: "+n);
}
}
class displaysum implements Runnable
{
int[] numbers; Thread t;
displaysum(int[] numbers)
{
this.numbers=numbers;
t=new Thread(this,"displaysum"); t.start();
}
```

```
public void run() { int sum=0;
for (int i = 0; i < numbers.length; i++) { sum+=numbers[i];
}
System.out.println("sum: "+sum);
}
}

public class a3q9 {
public static void main(String[] args) { Scanner sc=new Scanner(Sy
stem.in); int n=sc.nextInt();
new mainthread(n); sc.close();
}
}
```

```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac a3q9.java } ; if ($?) { java a3q9 }
6
cube: 250047
square: 1764
square: 484
cube: 250047
square: 2704
square: 1444
sum: 280
```

9) Write a multithreaded program that continuously prints the strings “OOPS” and “JAVA” in the console at random in the time elapse of 2 seconds. Use two threads one for “OOPS” another for “JAVA”

```
class thread1 extends Thread{ public void run(){
for(int i=0;i<10;i++){ System.out.println("JAVA");
}
}
}
class thread2 extends Thread{ public void run(){
for(int i=0;i<10;i++){ System.out.println("OOPS");
}
}
}
class a3q10{
public static void main(String args[]){ thread1 t1=new thread1();
thread2 t2=new thread2(); t1.start();;
```

```
t2.start();
}
}
```

```
PS C:\Users\91961\Documents\CODES>
                                  > cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac a3q10.java } ; if ($?) { java a3q10 }
JAVA
JAVA
JAVA
JAVA
JAVA
OOPS
OOPS
OOPS
OOPS
OOPS
OOPS
OOPS
OOPS
OOPS
OOPS
JAVA
JAVA
JAVA
JAVA
JAVA
```

10) Create a class called as clock which has three instance variables hours, minutes and seconds. It contains method called as add() which takes clock object as parameter and sets the data in the body and return the object. Now in main method create two clock objects and each one passes 3 arguments via parameterized constructor. Now create third object and this third object is assigned with adding the time values of each two objects

Like hint : Clock c3 = c1.add(c2)

Now after this print the addition of two time values

Testcases :

Time 1 : 5:23:12

Time 2: 6:38:50

Time after addition : 12:2:2

```
import java.util.*;
class clock{
int seconds,minutes,hours;
clock add(clock c2){
clock c3=new clock();
c3.seconds=this.seconds+ c2.seconds;
if(c3.seconds>60) {
c3.seconds = c3.seconds - 60;
c2.minutes++;
}
```

```
c3.minutes=this.minutes+ c2.minutes;
if(c3.minutes>60) {
c3.minutes = c3.minutes- 60;
c2.hours++;
}
c3.hours=this.hours+ c2.hours;
return c3;
}
}
public class a3q11 {
public static void main(String[] args) {
clock c1=new clock();
clock c2=new clock();
clock c3=new clock();
Scanner sc=new Scanner(System.in);
c1.hours=sc.nextInt();
c1.minutes=sc.nextInt();
c1.seconds=sc.nextInt();
c2.hours=sc.nextInt();
c2.minutes=sc.nextInt();
c2.seconds=sc.nextInt();
c3=c1.add(c2);
System.out.println(c3.hours+" "+c3.minutes+" "+c3.seconds);}}
```

```
PS C:\Users\91961\Documents\CODES> cd "c:\Users\91961\Documents\CODES\" ; if ($?) { javac a3q11.java } ; if ($?) { java a3q11 }
10 12 14
2 50 49
13 3 3
```