## Quiz-3(OOP with Java)

**Time: 30 Minutes** **Max Marks: 10**

**Multiple Choices [0.25X7 + 0.75X3]**

Q1. What is meant by time slicing or time sharing and Round robin scheduling in context of Multithreading?

Q2. What is the mechanism defined by java for the Resources to be used by only one Thread at a time?

Q3.What are all the four states associated in the thread?

Q4. You need to store elements in a collection that guarantees that no duplicates are stored and all elements can be accessed in natural order. Which interface provides that capability?

A.java.util.Map B.java.util.Set C.java.util.List D.java.util.Collection

Q5. Which of the following methods is a method of wrapper Integer for obtaining hash code for the invoking object?

a) int hash() b) int hashcode() c) int hashCode() d) Integer hashcode()

Q6. Which of these is a super class of wrappers Long, Character & Integer?

a) Long b) Digits c) Float d) Number

Q7. What will happen if two thread of same priority are called to be processed simultaneously?

a) Any one will be executed first lexographically b) Both of them will be executed simultaneously
c) None of them will be executed d) It is dependent on the operating system.

Q8. Which of these statements is incorrect?

a) By multithreading CPU's idle time is minimized, and we can take maximum use of it.

b) By multitasking CPU's idle time is minimized, and we can take maximum use of it.

c) Two thread in Java can have same priority

d) A thread can exist only in two states, running and blocked.

Q9. All collection classes are available in which package

A. java.io B. java.lang C. java.awt D. java.util

Q10. Which interface provides the capability to store objects using a key-value pair?

A. Java.util.Map B. Java.util.Set C. Java.util.List D. Java.util.Collection

## True/ False with appropriate reason[4X0.75]

Q1.The run() method should be overridden in classes created as subclass of thread?

Q2. The class Collection is the super class of HashMap class.

Q3. ArrayList collection class allows you to grow or shrink its size and provides indexed access to its elements.

Q4. An Iterable object can be used to traverse through a collection.

Q1. **Specify output [0.75]**

```
List words = new LinkedList();
int k;
for (k = 1; k <= 9; k++)
    words.add("word", k);

for (k = 1; k <= words.size(); k++)
    if (k % 3 == 0)
        words.remove(k);
System.out.println(words);
```

**Q2. Specify output [0.75]**

```
class Output {
      public static void main(String args[]) {
         Integer i = new Integer(257);
         byte x = i.byteValue();
         System.out.print(x);
      }
   }
```

**Q2. Specify output [0.75]**

```
List list = new LinkedList();
list.add("[");
 list.add("A");
 list.add("]");
System.out.println(list);

ListIterator it = list.listIterator();
while(it.hasNext()
{
   if ("[".equals(it.next())
||"]".equals(it.next()))
           it.remove();
    else
           it.add("*");
}
System.out.println(list);
```

**Q4. Specify output [0.75]**

```
Stack stk = new ArrayStack();
stk.push("A");
stk.push("B");
stk.push(stk);
while(!stk.isEmpty())
{   Object obj = stk.pop();
    if (obj instanceOf Stack)
    {
        while(!((Stack)obj).isEmpty())
        {
        System.out.print(((Stack)obj).pop());
    }
    else
        System.out.print(obj);
}
```