National University of Computer and Emerging Sciences, Lahore Campus

Course:	Advanced Programming	Course Code:	CS433
Program:	BS (Computer Science)	Semester:	Fall 2018
Duration:	60 Minutes	Total Marks:	23
Paper Date:	17-Nov-18	Weight	15 %
Section:	A, B, C	Page(s):	6
Exam:	Mid-II		

Section 1 (Objective part) [points 5]

Clearly circle the correct options.

Q1. Which statements are incorrect about wait(), notify() and notifyAll()?

[✓ (A) It is not important to acquire object lock before calling wait(), notify() and notifyAll().]

(B) Threads can communicate with each other by using wait(), notify() and notifyAll() methods.

5

(C) they are methods of Thread class (D) All are incorrect

Q2. What is a listener in context to event handling?

a) A listener is a variable that is notified when an event occurs.

[✓ b) A listener is a object that is notified when an event occurs.]

c) A listener is a method that is notified when an event occurs.

d) None of the mentioned

Q3. Which method is used to make main thread to wait for all child threads

[✓ a) Join ()] b) Sleep () c) Wait () d) Stop ()

Q4. Is it possible to convert a normal user thread into a daemon thread after it has been started?

(A) True [✓ (B) False] (C) depends on nature of the thread (D) depends on its parent thread

Q5. Serialization in RMI is

a) process of servicing RMI requests in an RMI server one at a time

[✓ b) conversion of Java datatypes into a sequence of bytes for communication across a network connection]

c) batching of RMI requests into series

d) None

Section 2 (Subjective part) (marks 14)

Question No. 1 [3 Marks]

The following code is part of airline booking system that is designed to be used in a single threaded application.

```
public class SeatCounter
{
      private int count = 0;

      public int getSeatCount() {
         return count;
      }

      public void bookSeat() {
         count++;
      }

      public void unBookSeat() {
         count--;
      }
}
```

b) Fix the code so that it works when used with multiple concurrent threads. [mark 1]

```
public class SeatCounter extends Thread
  ...

 synchronized void bookSeat(){
           count++;
 }

 synchronized void unbookSeat(){
           count--;
 }
```

1

a) Describe a potential problem with this code when used with multiple concurrent threads. Provide an example that demonstrates the problem. [marks 2]

when multiple concurrent threads are run using this code, the potential problem is the multiple thread can access the bookSeat and unBookSeat functions.

Example:

```
    public static void main(String args[]){
        SeatCounter obj1 =new SeatCounter();
        SeatCounter obj2 =new SeatCounter();
        obj1.bookSeat();
        obj2.unBookSeat();    }
```

2

No. 2 [3 Marks] Reg #: ______________ Section: __________

ollowing Java code executes a simple query. Fill in the blanks.

```
final int div_num = 113;
String sql = "SELECT div_name FROM division WHERE div_num = ?";
__class__.forName("com.mysql.jdbc.Driver");
__Connection__ conn =
DriverManager.getConnection("jdbc:mysql://localhost/mydb?"+
"user=root&password=root");
__PreparedStatement__ st = conn.prepareStatement(sql);
st.__set~~get~~Int__(1, div_num);
__ResultSet__ rs = st.executeQuery();
while (rs.next()) {
    String div_name = rs.getString(~~2~~)("div_name");
    System.out.println(div_name);
}
```

3

Question No. 3 [7 Marks] Short Questions

(i) What is the purpose of adapter classes like the MouseAdapter in Java? [1 mark]

Ans: The purpose of adpadter classes is to facilitate use by avoiding the implementation of all functions of MouseListener interface. MouseAdapter extends MouseListener.

1

(ii) Suppose you have 2 threads (Thread-1 on object1 and Thread-2 on object(2). Thread-1 is in static synchronized method1(), can Thread-2 enter static synchronized method2() at same time? Explain the reason. [1 mark]

Ans: No, because it is class level/Based synchronization, only one ~~one object of that class can call~~ thread can enter only one object's synchronized method at the same time.

1

(iii) Suppose you have thread and it is in synchronized method and now can thread c
static synchronized method from that method? Explain the reason. [1 mark]

Ans: Yes, We know that sychronized keyword is re-entrant. If we have acquired lock in a method and if that method calls another method that is also synchronized then we can enter in that second Method without acquiring lock

1

(iv) What is the output of the following code?

```
public class MultiThreading {

    private static class MyThread extends Thread {

        public MyThread(String name) {
            super(name);
        }
        public void run() {
            System.out.println(Thread.currentThread().getName());
        }
    }
    public static void main(String[] args) {
        MyThread myThread = new MyThread("myThread");
        myThread.run();
    }
}
```

Output:

main

(v) HashMap vs Hashtable : Suppose you are implementing a system and you want to choose one of them. [1 mark]

A) If your application is single-threaded, which one is preferable and Why?

1 HashMap. Because HashMap is not synchronized. As our application is single thread and we donot require synchronization.

B) If your application is multi-threaded, which one is preferable and Why?

Hashtable. Because Hashtable is synchronized. As our application is multi-threaded so we require synchronization to work our code correctly.

(vi) What is difference between PreparedStatement and CallableStatement? Which type of objects (data types) can be returned by PreparedStatement and CallableStatement? [2 marks]

PreparedStatement is type of database statement which is used to run queries we donot pass parameters directly but bind them later and then execute statement. on the other hand, CallableStatement is database statement that is used to execute stored procedures.

2 ResultSet can be returned by both of these statements depending upon query.

Reg #: ____________ Section: ____________

Question No. 4 [5 points] Export Objects over RMI: Suppose FAST is providing a remote service for admin to get the list of students registered in a particular course . The remote service consists of a method with the following signatures: **List<Strudent> getRegisteredStudents (String courseID).** Answer the following questions:

i) Which classes and interfaces will be implemented by the Service provider (Server)? [0.5 marks]

Student Class
Server class
ServiceInterface Interface
ServiceInterfaceImpl class

ii) Which classes and interfaces will be implemented by the admin (client)? [0.5 marks]

Student Class
Client class

iii) Which classes and interfaces will be given (shared) to the client by the server? [0.5 marks]

ServiceInterfaceImp_Stub
Service Interface Interfac
ServiceInterfaceImpl class

1.5

iv) Write down all the steps to create and run the application successfully. [1 mark]

i) set directories
ii) create class files of all java files on Server. (compile)
iii) compile all classes and files on client.
iv) rum rmic on SeviceInterfaceImp to create Stub.
v) rum rmicregistry server.
vi) run Server files
vii) run Client files

1

Name: ____________ Reg #: ____________

v) The ....ServiceInterface...... class is compiled by ..rmi........ compiler to gen

mark]

1

vi) What is the purpose of STUB? [1 mark]

The purpose of STUB is to provide communication medium to client to communicate with server. It is called as client interface.

1

vii) Write the necessary code for interface and implementation: [1.5 marks]

Ans (write necessary code for interface and implementation class here): Assume the data of students is stored in a HashMap.

Interface:

```
public interface ServiceInterface extends java.rmi.Remote {

    HashMap<Student> getRegisteredStudents(String CourseID)
        throws RemoteException;

}
```

Implementation

```
public interface ServiceInterfaceImpl extends java.rmi.Server.
    UnicastRemoteService implements ServiceInterface {
HashMap <Student> getRegisteredStudents (String CourseID) {

}
}
```

1.5