Fall Semester 2020-21
CSE1007 – Java Programming Lab
Practice Problems on Threads and Exceptions

1. Write a program to demonstrate the knowledge of students in Java Exception handling.
   Eg., Read the Register Number and Mobile Number of a student. If the Register Number does not contain exactly 9 characters or if the Mobile Number does not contain exactly 10 characters, throw an IllegalArgumentException. If the Mobile Number contains any character other than a digit, raise a NumberFormatException. If the Register Number contains any character other than digits and alphabets, throw a NoSuchElementException. If they are valid, print the message 'valid' else 'invalid'.

2. Write a program to demonstrate the knowledge of students in Java Exception handling.
   Create a class by name Employee with members – Employee ID, Name and year of birth. The Employee ID is a string that contains the ID in the format year-designation-number. The year is represented with the last two digits. The designation is a single letter code - 'F' for faculty and 'S' for staff. The number is a 3 digit number.
   (Example: 81-F-112   79-S-254)
   Write a program to read the employee details and validate the employee code. If the employee code is incorrect throw a user-defined exception "InvalidEmployeeCode" else create the Employee object and display the details of the employee.

3. Write a program to demonstrate the knowledge of students in multithreading.
   Eg., Three students A, B and C of B.Tech-IT II year contest for the PR election. With the total strength of 240 students in II year, simulate the vote casting by generating 240 random numbers (1 for student A, 2 for B and 3 for C) and store them in an array. Create four threads to equally share the task of counting the number of votes cast for all the three candidates. Use synchronized method or synchronized block to update the three count variables. The main thread should receive the final vote count for all three contestants and hence decide the PR based on the values received.

4. Create a User Defined Exception, Implement a remainder ClockAPP to the person saying Good Morning, Good Afternoon and Good Night by showing the current time. Write a code that assigns different values to the String variable timeofDay based on the value of time in hours. Raise remainder if time is 5 and 12, including 5 but not including 12, set timeofDay to "Morning, Have Fresh Vegetable Juice and then Sugar Tablet with mild walking". Raise Exception, if time is between 12 and 17, including 12 but not including 17, set timeofDay as "its Day Time, after lunch have tablet to avoid sleep". If time is between 17 and 18.30. Including 17 but not including 18.30, set timeofDays as "Hello, Good Evening have a

dinner" and finally, if time variable does not satisfy all the above condition, set time as "Night, Go for sleep".

5. Write a program to demonstrate the knowledge of students in File handling. Eg., Define a class 'Donor' to store the below mentioned details of a blood donor. Name, age, Address, Contactnumber, bloodgroup, date of last donation. Create 'n' objects of this class for all the regular donors at Vellore. Write these objects to a file. Read these objects from the file and display only those donors' details whose blood group is 'A+ve' and had not donated for the recent six months.