Ahmedabad University
School of Engineering and Applied Science
B. Tech. (ICT) Semester – II
CSC103 – Object Oriented Programming Lab
Lab Assignment – 1

Assignment Date: January 8, 2019 **Last Date of Submission: January 24, 2019**

Topics covered: Introduction to OOP and Java

Fundamentals of JAVA Programming, Java features, Process of Compilation and Execution, Java architecture, Environment and Tools, Data types, Basic syntax, Expressions, Control flow statements and Arrays.

Important Instructions:

Follow below coding and naming conventions while developing your programs:

1. Class names should be nouns and first letter must be capital letter. The first letter of each internal word capitalized. Use whole words-avoid acronyms and abbreviations E.g. BankAccount, Employee, AdminStaff, SavingAccount
2. Methods should be verbs with camel case, first letter lowercase and the first letter of each internal word capitalized.
   E.g. addEmployee( ); deleteEmployee( ); displayEmployeeList( );
3. Variables: Variable names should be meaningful. E.g. for Employee Class, variable names should be employeeId, employeeName, employeeDesignation, employeeScaleOfPay, etc.
4. Every Program should have header having following information in multi-line comments or java doc comments

```
/*
   @author RollNo. - Firstname Lastname                    @version dd/mm/yyyy
  Description: Write program definition.
*/
```

   Every Program should have footer with output of the Program in multi-line comment

```
/*  PROGRAM OUTPUT  */
```

Policy on Plagiarism (Copying of code/programs) and academic dishonesty

Copying Lab assignment/program code doesn't help you to learn the concepts.

Programs designed, developed and submitted by a student should be the result of original and individual work based on his/her own efforts. Full or part of the code should not be copied from internet or from peer students or other sources. A student should not share/circulate the code/programs developed by them (for individual assignments) with their peers in any form. Violation of above will be considered as academic dishonesty and any such case will be strictly dealt with and liable to get zero in the evaluation.

1. Declare data members of various data types like like short, int, long, float, double, char, String and boolean.
Also, use the data type casting and data conversion methods to convert float to int, double to float, double to float, String to int, int to String, String to float, float to String,
Print the values of various variables using overloaded System.out.print( ) and System.out.println( ) methods.

```
INTEGERS: whole valued signed numbers:
byte  :        1 byte : -128 to 127
short :        2 byte : -32768 to 32,767
int :          4 byte : 10 digits = -2,147,483,648 to                    2,147,483,647
long :         8 byte:19 digits = -9,223,372,036,854,775,808

FLOATING-POINT numbers: to represent decimal numbers (with fractional precision).
float          : 4 byte : = 3.4 e-38 to 3.4 e+38 (6-7 significant decimal digits)
double         : 8 byte : 1.7 e-308 to 1.7e+308  (15 significant decimal digits)
                                  to 9,223,372,036,854,775,807
char: 1 byte e.g. char='a';
boolean flag=true;

- To store more number of digits and higher precision, Java API has following classes:
- java.math.BigInteger
- java.math.BigDecimal
```

Literal Value	Data Type
178	int
8864L	int var1 = 8888864L;
37.266	double
87.363f	float f = 87.363;
26.77e3	double
'c'	char
true	boolean
false	boolean

2. User input demo - Write a program to read the student name (firstname <space> lastname), age and CGPA of a student from the user and display the details on the console. Display the age of the user in next year will be <ageNextYear>.

```
Scanner input = new Scanner(System.in);
String name = input.nextLine();
```

3. Write a program to find the maximum and minimum value entered by the user as command line arguments. Hint: int intValue = Integer.parseInt(args[i]);

4. **Loan payments –** Write a program that lets the user enter the interest rate, number of years and loan amount, and calculates the monthly and total payments.

$$monthlyPayment = \frac{loanAmount \times monthlyInterestRate}{1 - \frac{1}{(1 + monthlyInterestRate)^{numberOfYears \times 12}}}$$

$$totalPayment = monthlyPayment \times numberOfYears \times 12$$

Sample output:
Enter loan amount: 100000
Enter loan term (in years): 15
Enter interest rate: 10

Loan Amount: 100000
Loan Term: 15
Interest Rate: 10.00
Monthly Payment in INR is 1074.61
Total Payment in INR is 193428.92

5. **Math learning tool for first grader school student**
The program randomly generates 2 numbers and asks question such as "What is 9-2?". After the user enters the answer, program displays whether it is correct or not.

Hint: How to generate random numbers:

Method-1
The Math class contains the static Math.random() method to generate random numbers of the double type. The random() method returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.

```
int number1 = (int)(Math.random() * 10)      //this generates single digit integer
```

Method-2

```
import java.util.Random;
Random rand = new Random();
// Generate random integers in range 0 to 999
int rand_int1 = rand.nextInt(1000);
```

Extend the above program using while loop to generate 5 questions repeatedly. After a student answers all five, report the number of correct answers. Also, display the time spent on the test.
Hint:

```
long startTime = System.currentTimeMillis( ); //current system time in milli-seconds
```

6. Lottery simulation using random number

Generates a random two-digit number string, prompt the user to enter a two-digit number, and determines whether the user wins according to the following rule:

1) If the user input matches the lottery number in the exact order, the award is $10,000.
2) If all the digits in the user input match all the digits in the lottery number, the award is $3,000.
3) If one digit in the user input matches a digit in the lottery number, the award is $1,000.
4) Else, if no digit matches, display “Sorry! Better luck next time”.

You may get the solution without using String and using String:

```
System.out.print("Enter your lottery pick (two digits): ");
int guess = input.nextInt();

// Get digits from lottery
int lotteryDigit1 = lottery / 10;
int lotteryDigit2 = lottery % 10;

// Get digits from guess
int guessDigit1 = guess / 10;
int guessDigit2 = guess % 10;
```

```
Hint: Below is one possible solution using String
// generate a lottery as a two-digit string
String lottery = "" + (int)(Math.random() * 100);

System.out.print("Enter your lottery pick (two digits): ");
String guess = input.nextLine();

// Get digits from lottery
char lotteryDigit1 = lottery.charAt(0);
char lotteryDigit2 = lottery.charAt(1);

// Get digits from guess
char guessDigit1 = guess.charAt(0);
char guessDigit2 = guess.charAt(1);
```

7. Number guessing game:

Write a program that randomly generates an integer between 0 and 100 inclusive. Prompt the user to enter a number continuously (until 5 or 10 guesses) until the number matches the computer generated random number. For each user input, the program should also tell the user, whether the input is too low or too high.

int randomNumber = (int)Math.random( ) * 100;

8. Write a program to create and display unique three-digit number using 1, 2, 3, 4. Also count how many three-digit numbers are there.

9. Write a program to rotate an array (of any length) of integers in left direction.
Test Data: {20, 30, 40}
Expected output: {30, 40, 20}
Hint: Method to print an array
System.out.println("Original Array: " + java.util.Arrays.toString(intArray));

10. Write a program to rearrange all the elements of a given array of integers so that all the odd numbers come before all the even numbers.
Original Array: [1, 7, 8, 5, 7, 13, 0, 2, 4, 9]
New Array: [7, 5, 7, 13, 9, 1, 0, 2, 4, 8]

11. Marksheet Generation using Arrays:
List of roll numbers of 10 students and their marks in a semester out of 700 are given. Calculate their percentage and assign them grades according to following criteria:

Percent	Grade
70 to 100	DIST
60 to 69.99	FIRST
50 to 59.99	SECOND
35 to 49.99	THIRD
0 to 34.99	FAIL

The program should display the result in 3 columns: rollNo, percent, Grade for 10 students.

ROLLNO	MARKS	PERCENTAGE	GRADE
1	500	71.43	DIST
2	550	78.57	DIST
3	200	28.57	FAIL
4	350	50.00	SECOND
5	680	97.14	DIST

(**HINT**:
Take 2 arrays one for rollNo and other for Marks.
Calculate the percentage from marks and store it in another array.
Apply the Grading logic on Percentage array, and store the grades in another array.)

Syntax to declare arrays in Java:
//array creation and initialization
int marks[ ] = {600, 500, 400, 350, 450, 650};

//array declaration
int percent[ ] = new int[10];

```
//array initialization
percent[0] = 75.50;
```

12. **Retirement Planning Program** – The program should ask the user for following inputs:
Money that the user needs on retirement.
How much money the user wants to contribute every year towards retirement fund?
Interest rate (in %)
After taking above inputs from the user, the program calculates the number of years in which the user can retire from his job/occupation.

```
Hint: use while loop
while(balance < goal)
double interest = balance * interestRate / 100;
balance += interest;
year++;
```

13. Salesman salary in department store consists of base salary and a commission. The base salary is INR 5000. The below scheme is used to determine the commission rate.

Sales Amount (in INR)	**Commission Rate**
0-5,000	8 %
5000.01 – 10,000	10 %
10,000.01 and above	12 %

Find the minimum sales amount that is needed by Salesman to earn INR 25,000 as commission using do-while loop.

14. Write an interactive program to print a diamond shape. For example, if user enters the number 3, the diamond will be as follows:

```
        *
    *       *
*       *       *
    *       *
        *
```

15. Accept a line and check how many consonants and vowels are there in line.

16. Write a program to ask names of 5 cities from the user and perform alphabetic sorting on the names of cities. You may use any sorting algorithm.

Hint: for comparing 2 strings, use the below method of **String** class:

```
if(s1.compareTo(s2) > 0) {
    //swap s1 with s2 in the string array
}
```

17. Write a program to display login form which asks username and password from the user.
The program takes username and password from the JFrame GUI/console and validates it with the list of valid usernames and password (use two arrays of valid usernames and password) stored in the program (class data members/instance variables) and display “successful login” or “invalid

username and password" message. On successful login, redirect the user to the welcome screen (WelcomeJFrame).

Validation: check that username and password must be minimum 6 chars in length.

18. Write a program to capitalize the first letter of each word in a sentence entered by the user.

19. Write a program to check if a given string has all unique characters.

```
char[] chars = str.toCharArray();
Arrays.sort(chars);
```

20. Take date of birth from the user and calculate the age of the person in years, month and days as per the below screen GUI design.

You can use the new Java 8 Date/Time API

```
import java.time.*;

LocalDate today = LocalDate.now();
LocalDate birthday = LocalDate.of(yy, mm + 1, dd);
Period p = Period.between(birthday, today);
System.out.println(p.getYears());
System.out.println(p.getMonths());
System.out.println(p.getDays());
```

Additional Practice Questions

Note: The below questions are for extra practice and revision and NOT required to be submitted as part of an assignment.

1. Write a Java program to check whether given number is odd or even.
2. Write a Java program that takes three numbers as input to calculate and print the average of the numbers.
3. Take 10 integer values from the user and display the min, max and average value. [Use of Array is better option]
4. Program to calculate Simple Interest for the given principal amount, rate of interest and no. of years.
5. Declare two arrays with sample values using array initialization list. Create a derived third array whose each element should be equal to element of first array plus element of second array

   Third Array first element = {first element of first array} + {first element of second array}

   Third Array second element = {second element of first array} + {second element of second array}

   …….
6. Write a program to take a string as input from the user and count the number of alphabets, digits, lower case letters, upper case letters, white space letters in the output.

   Hint: Character.isLetter( ), Character.isDigit( ), Character.isSpaceChar( )
7. Accept one or two string from the user as per the below operations and perform following operations on string using a menu driven program:
   a) Get a string from user and display length of that string.
   b) Get two strings from user and compare weather both are same or not.
   c) Get a string from user and display reverse of it.
   d) Get two strings from user and concatenate the second string at end of first string.
   e) Check whether the string is palindrome
   f) Search for a sub-string within the original string
8. Write a program to count the number of words that start with capital letters.
9. Write a Java program to convert an octal number to a hexadecimal number.
10. Write a Java program to count the letters, spaces, numbers and other characters of an input string.

11. Write a Java program to calculate the yearly depreciation of an asset using the SLM (Straight line method) formula:

    Yearly Depreciation = (Purchase price of Asset – scrap value) / life of asset

    Take Purchase price, scrap value and life of asset in years from the user.

12. Write a Java program to check whether Java is installed on your computer.

13. Write a compound interest based program to find the maturity amount of fixed deposit based on the principal amount, rate of interest and number of years entered by the user.

    Display the principal amount, rate, years and maturity Amount on the console.

    Use the Compound Interest Formula **MaturityAmount = P (1+r/100)^n**

    P = principal amount, r =rate of interst, n=number of years.

    (**HINT**:Use z = Math.pow(x,y); use float or double type variables

    (Math is the inbuilt-class where the power method is defined.)

    Display the formatted maturity amount 2 digits after decimal point using

    System.out.printf("Maturity Amount = %.2f", maturityAmount);

14. Write a Java program to convert a hexadecimal to a decimal number.

    Hint: Input a hexadecimal number: 25

    Expected Output: Equivalent decimal = 37

15. Write a Java program to convert a hexadecimal to a binary number.

    Hint: Enter Hexadecimal Number : 7

    Expected Output: Equivalent Binary = 111

16. Write a Java program and compute the sum of the digits of an integer.

    Input Data: Input an integer: 51

    Expected Output: Sum of the digits = 6

In case of any doubt, contact course faculty: Prof. Kuntal Patel.

All the Best….

* * * * *