Assignment 1 Algorithms Answers

1. Check if the given number is EVEN or ODD.

1) Take input from user store its value in variable num
2) If (num % 2 ==0) then print the number is EVEN
3) Else print the number is ODD
4) End the program

2. Write a Java Program to find the Factorial of a given number

1) Take the input from user and store its value in a variable named num
2) Declare a integer variable result and assign it the value 1
3) If num is equal to 0 or 1, then print the result 1
4) Else
    a. For i=1 to n
        result = result * i
    b. End for
5) Print the result
6) End the program

3. Find the Factorial of a number using Recursion.

1) Take input from user and store its value in variable in num
2) Declare a long integer variable ‘factorial’
3) Call the multiplyNumbers method with num as the input parameter and assign the result to factorial.
4) Print the factorialon
5) Declare the multiplyNumbers method that takes an integer parameter num and returns a long.
6) Inside the multiplyNumbers method, check if num is greater than or equal to 1.
7) If num is greater than or equal to 1, multiply num by the result of calling multiplyNumbers with the parameter num-1
8) If num is less than 1, return 1
9) Return the result of the multiplication performed in step 9.
10) End the program

4. Swap two numbers without using the third variable approach.

1) Take two numbers as input and store its value in variable a & b.
2) Add the value of a & b and store the result of addition in variable a.
3) Subtract the value of b from the new value of variable a and store the result in variable b
4) Subtract the value of b from the new value of variable a and store the result in variable a
5) Print the new value of a & b

```
6) End the program
```

5. How to check whether the given number is Positive or Negative in Java?

```
1) Take input from user and store its value in variable num
2) If the number is greater than or equal 0 then print “Number is Positive”
3) Else print “Number is Negative”
4) End the program
```

6. Write a Java Program to find whether a given number is Leap year or NOT

```
1) Take an input from the user and store its value in year variable.
2) Initialize a boolean variable leap to false.
3) Check if the year is divisible by 4.
      i. If it is, check if the year is divisible by 100.
            1. If it is, check if the year is divisible by 400.
                  a. If it is, set leap to true.
                  b. Otherwise, set leap to false.
            2. Otherwise, set leap to true.
     ii. Otherwise, set leap to false.
4) Check the value of leap.
      i. If it is true, print that the year is a leap year.
     ii. Otherwise, print that the year is not a leap year.
5) End the program
```

7. Write a Java Program to Print 1 To 10 Without Using Loop.

```
1) Define a method called “printNumbers” that takes an integer  as an
   argument.
2) If n is less than or equal to 10, then do the following:
      i. Print the value of n
     ii. Call the “printNumbers” method with ‘n+1’ as the arguments
3) In the main method call the “printNumbers” method with 1 as the argument
4) End the program
```

8. Write a Java Program to print the digits of a Given Number

```
1) Take input from user and store its value in variable num
2) While num is greater than 0
      i. Calculate digit = num%10 to extract the last digit of num
     ii. Print digit
    iii. Update num by dividing it by 10, num /=10
3) End the program
```

9. Write a Java Program to print all the Factors of the Given number

1) Take input from user and store its value in variable ‘number’
2) Start a for loop with condition i<=number, incrementing i by 1
3) Within the loop, check if number is divisible by i using the modulus operator (%). If the result is 0, then i is a factor of number.
4) If i is a factor of number, print the value of i followed by a space.
5) At the end of loop all the factor of ‘number’ would have been printed
6) End the program

10. Write a Java Program to find the sum of the digits of a given number.

1) Take input from user and store its value in integer variable ‘num’
2) Initialize a variable sum to 0 to store the sum of the digits
3) Set up a While loop with condition num is not equal to 0
   i. Add the last digit of num (obtained using num%10) to sum
   ii. Remove the last digit from num by dividing it by 10
4) Print the value of num
5) End the program

11. Write a Java Program to find the smallest of 3 numbers (a,b,c)

1) Take 3 numbers input from user and store its value in variables a, b, c
2) Initialize a variable “smallest” with the value of ‘a’
3) Check if ‘b’ is smaller than ‘smallest’. If it is assign ‘b’ to smallest
4) Check if ‘c’ is smaller than ‘smallest’. If it is assign ‘c’ to smallest
5) Print the value of smallest variable
6) End the program

12. How to add two numbers without using the arithmetic operators in Java?

1) Take 2 input from the user and store its value in variable a & b
2) Create method add which takes a and b variables as parameters.
3) While the value of ‘b’ is not zero, do the following
   i. Create a variable named ‘carry’ and set it to the result of the bitwise AND operation between ‘a’ & ‘b’
   ii. Update the value of "a" to the result of the bitwise XOR operation between "a" and "b".
   iii. Update the value of "b" to the result of left shift "carry" by 1.
4) Return "a" as the final result of the function.
5) Print the result
6) End the program

13.Write a java program to Reverse a given number.

1) Take input from user and store its value in integer variable ‘num’
2) Create a variable result to store the reserved number and initialize it to 0
3) While the input num is greater than 0, do the following
    i. Create a variable ‘digit’ and assign it value of num%10
    ii. Assign the value (result*10+digit) to result
    iii. Assign the value (num/10) to variable num
4) Print the result
5) End the program

14.Write a Java Program to find the GCD of two given numbers.

1) Take two input numbers as num1 and num2.
2) Check if num1 or num2 is zero, if any of them is zero then return the other number as GCD.
3) If both num1 and num2 are not zero, divide the larger number by the smaller number and get the remainder.
4) Now, set the larger number to be the smaller number and the remainder to be the larger number
5) Repeat step 3 and step 4 until the remainder becomes zero.
6) The GCD is the last non-zero remainder obtained in step 3.
7) Print the result
8) End the Program

15.Write a java program to LCM of TWO given numbers

1) Take two inputs from the user and store its value to n1 and n2
2) Initialize the integer lcm with the maximum value between n1 and n2
3) Start an infinite loop with the while(true) statement.
4) Check if lcm is divisible by both n1 and n2 using the modulo operator (%). If lcm % n1 == 0 and lcm % n2 == 0, proceed to step 5. Otherwise, proceed to step 6.
5) Print a message to the console that displays the values of n1, n2, and lcm and indicates that lcm is the LCM of n1 and n2. Then, exit the loop using the break statement.
6) Increment lcm by one using the ++lcm statement.
7) Repeat steps 4-6 until lcm is found to be the LCM of n1 and n2.
8) End the program.

16. **Write a java program to LCM of TWO given numbers using the Prime Factors method**

    1) Declare variables a, b, max, step, and lcm as integers and initialize lcm to zero.
    2) Take 2 input from the user and store its value in variable a & b
    3) Determine the maximum value of a and b using an if-else statement and set max and step to the maximum value
    4) While loop begins with the condition a!=0, do the following steps:
        i. Check if max is divisible by both a and b (max % a == 0 && max % b == 0).
        ii. If true, set the value of lcm to max and break out of the loop.
        iii. If false, increment max by the value of step.
    5) Print LCM
    6) End the program

17. **Check whether the Given Number is a Palindrome or NOT.**

    1) Initialize integer variables result, sum, and temp to 0
    2) Take input from user and store its value to variable n
    3) Set temp = n
    4) While n is greater than 0, do the following:
        i. Compute the remainder of n when divided by 10 and store it in result
        ii. Multiply sum by 10 and add r to it
        iii. Divide n by 10 and discard the remainder
    5) If temp is equal to sum, print "palindrome number"
    6) Otherwise, print "not palindrome"
    7) End the Program

18. **Write a Java Program to print all the Prime Factors of the Given Number.**

    1) Initialize a while loop to check for factor of 2
        i. While n is divisible by 2
        ii. Print 2 as a factor of n
        iii. Divide n by 2
    2) Initialize a for loop to check for odd factors up to the square root of ‘n’
        i. Set the loop variable ‘i’ to 3
        ii. While ‘i’ is less than or equal to the square root of ‘n’
        iii. If ‘n’ is divisible by ‘i’
        iv. Print ‘i’ as a factor of ‘n’
        v. Divide ‘n’ by ‘i’
        vi. Else, increment ‘i’ by 2
    3) If n is greater than 2, it is a prime number and should be printed as a factor of n.
    4) End the program

19.To print the following series EVEN number Series 2 4 6 8 10 12 14 16 .....

1) Take input from user upto which the user want the series to continue and store the value in variable number
2) Use a for loop to iterate over numbers from 1 to number, with loop variable i incrementing by 1 every time
3) For each iteration of the loop, check if the value of ‘i’ is even or not by this (i % 2 == 0)
4) If ‘i’ is even then print “i”
5) Else ‘i’ is odd, continue with next iteration until the number is reached
6) End the program

20.To print the following series ODD number Series 1 3 5 7 9 11 13..

1) Take input from user upto which the user want the series to continue and store the value in variable number
2) Use a for loop to iterate over numbers from 1 to number, with loop variable i incrementing by 1 every time
3) For each iteration of the loop, check if the value of ‘i’ is even or not by this (i % 2 != 0)
4) If ‘i’ is Odd then print ‘i’
5) Else ‘i’ is even, continue with next iteration until the number is reached
6) End the program