Group Project Work I

Objectives

After successful completion of the project, the students will be able to:

No.	Objectives	Mapping to CLO
1.1	Demonstrate problem solving skills by analyzing the given problem description using flow chart and pseudo code.	CLO-2
1.2	Design, write and debug a small-scale computer program by using the basic programming constructor's and applying good practices.	CLO-3, CLO-4
1.3	Use modern development and collaboration tools and techniques	CLO-2, CLO-3, CLO-4
1.4	Demonstrate teamwork, collaboration and presentation skills	Transferable Skills

General Direction

- The project team should use collaborative tools: ***Google doc*** for documentation and ***Git Hub*** for team collaboration on the implementation.
- Prepare a problem analysis and algorithm design document using google doc with proper cover page which includes group name, team member lists [name and Id].
- Your documentation should be well organized, formatted with (font, color, font family, spacing, size, etc.) and shows consistence.
- You will be required to use all topics that you've learnt so far. There will be a bonus if you use pointers.
- Identify and use appropriate programming constructor's and techniques.
- The implementation should satisfy the principle of writability/readability showing that you followed the best practices of program writing.
- Create a directory with name 'project-work' inside your group repository and allow team members to contribute to the implementation.
- The submission of all the project works is through Google Classroom by the team leader. She/he expected to provide implementation repository link and google doc link with proper permission.
- Submission deadline: April 10/2023.
- There will be a Q and A session on April 11, 2023.

Note:

- You can add any appropriate service/style /creativity of your own within your program

Part I: Problem Analysis, Program Design and Implementation

Instruction: Analyze the problem given below, and (b) draw ***flow chart*** and write a ***pseudo-code*** to describe the algorithm that you apply to solve the following problem and then transform the algorithms into ***program***.

Problem Descriptions

1.

It is difficult to make a budget that spans several years, because prices are not stable. If your company needs 200 pencils per year, you cannot simply use this year's price as the cost of pencils 2 years from now. Because of inflation the cost is likely to be higher than it is today. Write a program to gauge the expected cost of an item in a specified number of years. The program asks for the cost of the item, the number of years from now that the item will be purchased, and the rate of inflation as a percentage, like 5.6 (percent). Your program should then convert the percent to a fraction, like 0.056, and should use a loop to estimate the price adjusted for inflation. The program then outputs the estimated cost of the item after the specified period. The program should also display the cost difference between every year in tabular until the specified period. But allow the user to choose which price of the pencil he/she want to see (either the pencil cost of specific year from now on wards or each year of specified number of years in tabular format.)

2.

A company wants to transmit data over the telephone, but they are concerned that their phones may be tapped. All of their data are transmitted as four-digit integers. They have asked you to write a program that encrypts their data so that it can be transmitted more securely. Your program should read a four-digit integer and encrypt it as follows: Replace each digit by *(the sum of that digit plus 7) modulus 10*. Then, swap the first digit with the third, swap the second digit with the fourth and print the encrypted integer and vice versa to decrypt the number. Write a program that inputs four-digit integer and either encrypt to secrecy the inputted number or

decrypt it to form the original number. The program should allow the user to choose among encryption and decryption operation.

3.

A right triangle can have sides whose lengths are all integers. The set of three integer values for the lengths of the sides of a right triangle is called a Pythagorean triple. The lengths of the three sides must satisfy the relationship that the sum of the squares of two of the sides is equal to the square of the hypotenuse. Write an application that displays a table of the Pythagorean triples for **side1**, **side2** and the **hypotenuse**, all no larger than 100. Use a **triple-nested for loop** that tries all possibilities. This method is an example of "**brute-force**" computing.

4.

Write a program that convert decimal number to binary, octal and hexadecimal and vice versa. The program should accept an integer number and create number system conversion table. And also the program let the user to input the number in any format (i.e. in decimal or binary or octal or hexadecimal) and convert to equivalent all other number system.

5.

Interest on a loan is paid on a declining loan balance, and hence a loan with an interest rate of, say, 14 percent can cost significantly less than 14 percent of the loan balance. Write a program that takes a loan amount and interest rate as input and then outputs the monthly payments and balance of the loan until the loan is paid off. Assume that the monthly payments are one twentieth (1/20) of the original loan amount including monthly interest paid. The monthly interest payment is computed as *(remaining loan balance * interest rate)/12*, and that any amount in excess of the interest is credited toward decreasing the balance due. Also have the program output the total interest paid over the life of the loan in table format. Finally, determine what simple annualized percentage of the original loan balance was paid in interest.

For example, on a loan of 20,000 ETB, the payments would be 1,000 ETB a month. Assume if the interest rate is 10 percent, then each month the interest is one-twelfth of 10 percent of the remaining balance. The first month, (10 percent of 20,000 ETB)/12, or $166.67, would be paid

in interest, and the remaining 833.33ETB would decrease the balance to 19,166.67ETB. The following month the interest would be (10 percent of 19,166.67 ETB)/12, and so forth.

6.

The keypad on your oven is used to enter the desired baking temperature and is arranged like the digits on a phone as shown on the figure below. Unfortunately the circuitry is damaged and the digits in the leftmost column no longer function. In other words, the digits 1, 4, and 7 do not work. If a recipe calls for a temperature that can't be entered, then you would like to substitute a temperature that can be entered. Write a program that inputs a desired temperature. The temperature must be between 0 and 999 degrees. If the desired temperature does not contain 1, 4, or 7, then output the desired temperature. Otherwise, compute the next largest and the next smallest temperature that does not contain 1, 4, or 7 and output both. For example, if the desired temperature is 450, then the program should output 399 and 500. Similarly, if the desired temperature is 375, then the program should output 380 and 369.

1	2	3
4	5	6
7	8	9
	0	

7.

Write a C++ application that read a date as integer and extracts a day, month and year and determine whether the date is valid. If the program is given a valid date, an appropriate message is displayed. If instead the program is given an invalid date, an explanatory message is given. To recognize whether the date is valid, you must be able to determine whether the year is a leap year or not. If the given year is leaping year, then program prints the next 20 leap years otherwise a message that says "A valid date but not a Leap Year". Your program should also consider that century year cannot be leap year unless it is divisible by 400. See the example below for the expected input/output behavior for a valid date.

Please enter a date (dd mm yyyy): 3042006

30/4/2006 is a valid date but not a Leap Year

Please enter a date (dd mm yyyy): 1132006

Invalid month: 13

Please enter a date (dd mm yyyy): 2921899

Invalid day of month 29

Please enter a date (dd mm yyyy): 2922020

29/2/2020 is a valid date and also a Leap Year

Here is below the next twenty Leap years are:

"List of leap years separated by comma"

8.

Kokebe Tsibah secondary and preparatory prepare school closing function (gathering) and invite various concerned body. The people who attend the school function categorized in to four groups and given a code. Accordingly, 1 represents an infant, 2 represents a child, 3 represents a teenager, and 4 represents an adult. No other integer value should be accepted as valid input, and data entry should stop when a negative value is entered. Write an application that read entrance ticket number, person code as 1, 2, 3 or 4, and entrance fee. The fee schedule is 5ETB for child, 10ETB infant, and 20ETB for teenager and adults. The program should compute and display total numbers of people attend the function and total amount of ticket sales. Also the program should display a list of how many infants, children, teenagers, and adults were at the school function along with category codes. You should write the program without using array.

9.

A prime number is an integer greater than one and divisible only by itself and one. Write an application that reads an integer number N and check if it is prime number or not. If the number is prime then the program prints out the average value of prime numbers between 1 and N. Otherwise the program should find and prints out all prime factors of inputted number N. (Hint: 1 is a prime number. For each number from greater than or equal to 2, find *Remainder = N % k*, where k ranges from 2 to *sqrt(N)*. If k is greater than *sqrt(number)*, the number is not equally divisible by k. And if any *Remainder* equals 0, the number is not a prime number.)

10.

For your 21st birthday, your grandmother opens a savings account for you and deposits 1000 ETB into the account. The savings account pays a 5% interest on the account balance. If you don't deposit any more money into the account, and you don't withdraw any money from the account, how much will your savings account be worth at the end of through N years? Also consider the case that your mother deposits some amount of money every year and find your

savings account balance after specified number of years. Create a program that allows the user to enter the principal, the minimum and maximum interest rates, and the number of years, annual deposit amount your mother make and compute your account balance after specified number of years in both cases.

Given a formula: b = p * (1 + r) n.

Were, ***p*** is the principal (the amount of the deposit made),

r is the annual interest rate, ***n*** is the number of years, and

b is the balance in the savings account at the end of the nth year

PART II: Pattern Printing

Write a program that accepts a positive integer, *n*, or a letter entered by the user and prints the following shape. But if the value entered is less than one, the program prints nothing.

a)

```
0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 8   8 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7       7 6 5 4 3 2 1 0
0 1 2 3 4 5 6           6 5 4 3 2 1 0
0 1 2 3 4 5               5 4 3 2 1 0
0 1 2 3 4                   4 3 2 1 0
0 1 2 3                       3 2 1 0
0 1 2                           2 1 0
0 1                               1 0
0                                   0
0 1                               1 0
0 1 2                           2 1 0
0 1 2 3                       3 2 1 0
0 1 2 3 4                   4 3 2 1 0
0 1 2 3 4 5               5 4 3 2 1 0
0 1 2 3 4 5 6           6 5 4 3 2 1 0
0 1 2 3 4 5 6 7       7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 8   8 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0
```

b)

```
0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0
  0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 0
    0 1 2 3 4 5 6 7 6 5 4 3 2 1 0
      0 1 2 3 4 5 6 5 4 3 2 1 0
        0 1 2 3 4 5 4 3 2 1 0
          0 1 2 3 4 3 2 1 0
            0 1 2 3 2 1 0
              0 1 2 1 0
                0 1 0
                  0
                0 1 0
              0 1 2 1 0
            0 1 2 3 2 1 0
          0 1 2 3 4 3 2 1 0
        0 1 2 3 4 5 4 3 2 1 0
      0 1 2 3 4 5 6 5 4 3 2 1 0
    0 1 2 3 4 5 6 7 6 5 4 3 2 1 0
  0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0
```

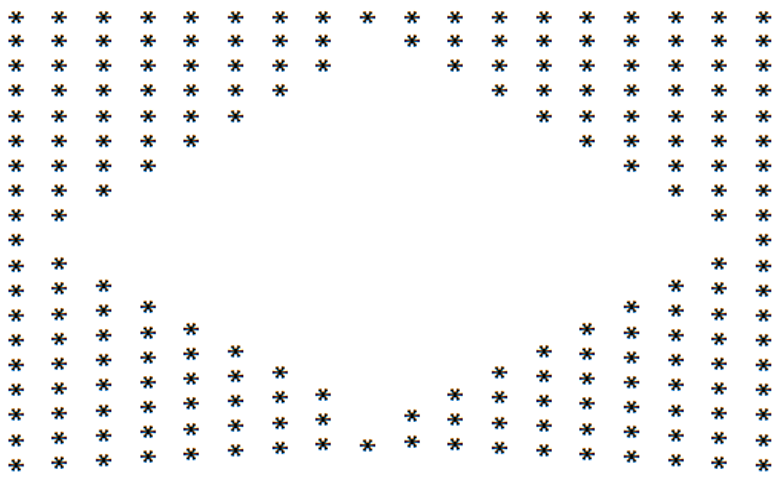
c)

d)

```
0                       0
10                     01
210                   012
3210                 0123
43210               01234
543210             012345
6543210           0123456
76543210         01234567
876543210     0123456789
98765432101234567890
876543210     0123456789
76543210         01234567
6543210           0123456
543210             012345
43210               01234
3210                 0123
210                   012
10                     01
0                       0
```

```
A B C D E F G H G F E D C B A
  A B C D E F G F E D C B A
    A B C D E F E D C B A
      A B C D E D C B A
        A B C D C B A
          A B C B A
            A B A
              A
            A B A
          A B C B A
        A B C D C B A
      A B C D E D C B A
    A B C D E F E D C B A
  A B C D E F G F E D C B A
A B C D E F G H G F E D C B A
```

e)



f)

g)

```
A B C D E F G H G F E D C B A
A B C D E F G   G F E D C B A
A B C D E F       F E D C B A
A B C D E           E D C B A
A B C D               D C B A
A B C                   C B A
A B                       B A
A                           A
A B                       B A
A B C                   C B A
A B C D               D C B A
A B C D E           E D C B A
A B C D E F       F E D C B A
A B C D E F G   G F E D C B A
A B C D E F G H G F E D C B A
```

```
9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9
  8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8
    7 6 5 4 3 2 1 0 1 2 3 4 5 6 7
      6 5 4 3 2 1 0 1 2 3 4 5 6
        5 4 3 2 1 0 1 2 3 4 5
          4 3 2 1 0 1 2 3 4
            3 2 1 0 1 2 3
              2 1 0 1 2
                1 0 1
                  0
                1 0 1
              2 1 0 1 2
            3 2 1 0 1 2 3
          4 3 2 1 0 1 2 3 4
        5 4 3 2 1 0 1 2 3 4 5
      6 5 4 3 2 1 0 1 2 3 4 5 6
    7 6 5 4 3 2 1 0 1 2 3 4 5 6 7
  8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8
9 8 7 6 5 4 3 2 1 0 1 2 3 4 5 6 7 8 9
```

h)

```
0                       0
0 1                   1 0
0 1 2               2 1 0
0 1 2 3           3 2 1 0
0 1 2 3 4       4 3 2 1 0
0 1 2 3 4 5   5 4 3 2 1 0
0 1 2 3 4 5 6 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 5 4 3 2 1 0
0 1 2 3 4 5 4 3 2 1 0
0 1 2 3 4 3 2 1 0
0 1 2 3 2 1 0
0 1 2 1 0
0 1 0
0 0
```

i)

```
H G F E D C B A B C D E F G H
  G F E D C B A B C D E F G
    F E D C B A B C D E F
      E D C B A B C D E
        D C B A B C D
          C B A B C
            B A B
              A
            B A B
          C B A B C
        D C B A B C D
      E D C B A B C D E
    F E D C B A B C D E F
  G F E D C B A B C D E F G
H G F E D C B A B C D E F G H
```

j)

```
A                             A
A B                         B A
A B C                     C B A
A B C D                 D C B A
A B C D E             E D C B A
A B C D E F         F E D C B A
A B C D E F G     G F E D C B A
A B C D E F G H H G F E D C B A
A B C D E F G H J H G F E D C B A
A B C D E F G H H G F E D C B A
A B C D E F G     G F E D C B A
A B C D E F         F E D C B A
A B C D E             E D C B A
A B C D                 D C B A
A B C                     C B A
A B                         B A
A                             A
```