

BlueCrest University College Monrovia
Department of OpenLabs
Software Engineering
Course: Android Programming Assignment.

Q1. Using Android programming technology, Create a flowchart and pseudocode for the following. The answer must include a flowchart, pseudocode, and program code as the solution to this question:

- i. Input a year and find whether it is a leap year or not.
- ii. Take two numbers and print the sum of both.
- iii. Take a number as input and print the multiplication table for it.
- iv. Take 2 numbers as inputs and find their HCF and LCM.
- v. Keep taking numbers as inputs till the user enters 'X', after that print the sum of all.

To solve this problem using Android programming, we'll break it down into the following components:

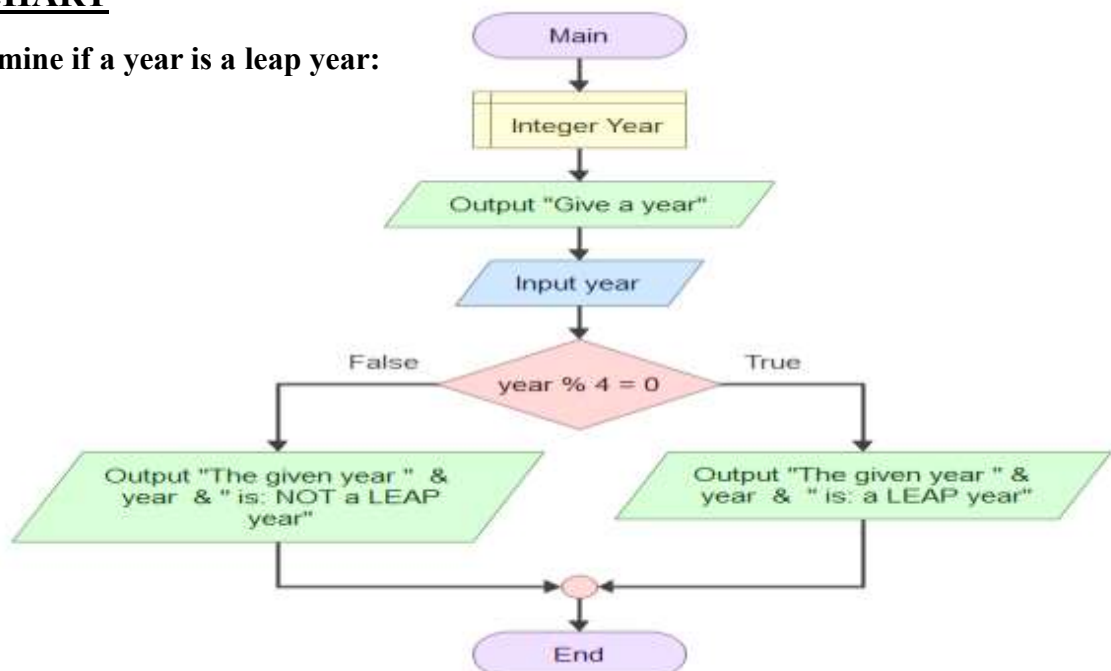
Flowchart: This will Visually represent the logical flow of the solution.

Pseudocode: The pseudocode will provide a structured and easy-to-understand solution outline in plain language.

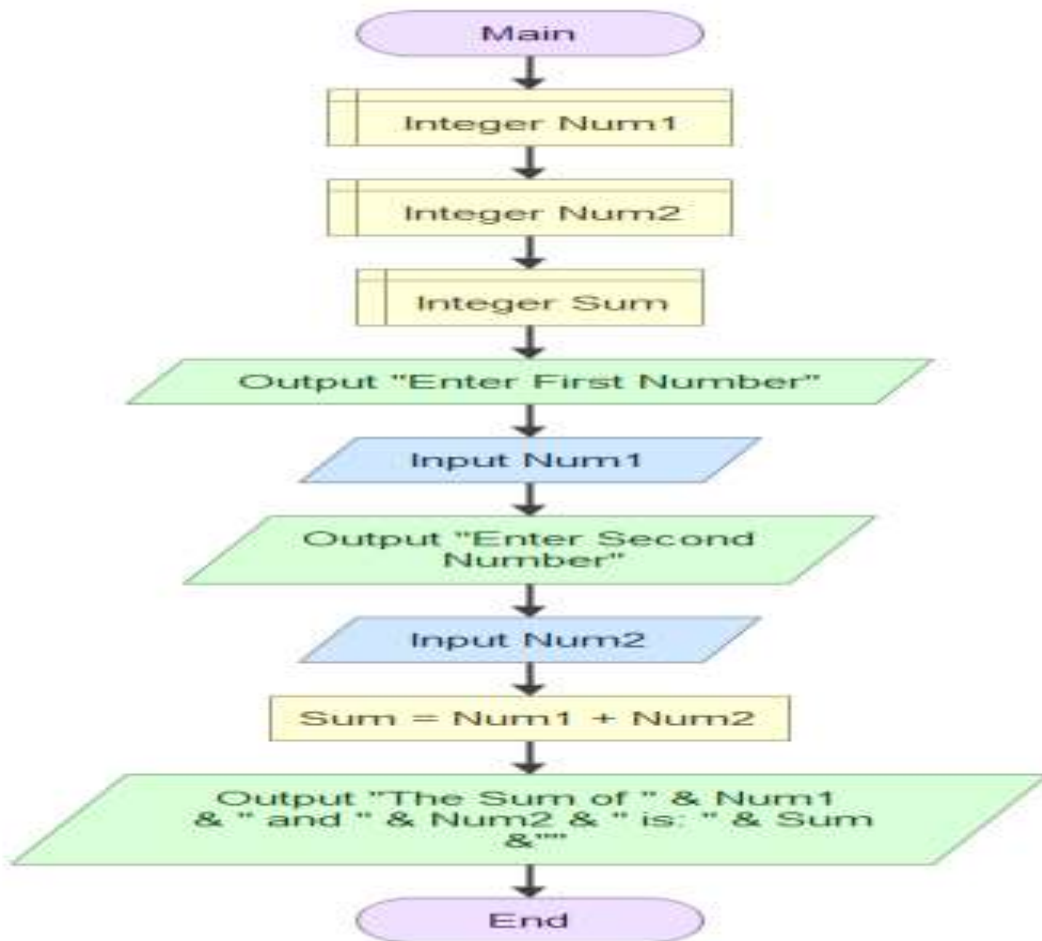
Program Code preferably Java for me: Finally, the Android program code in Java will implement the solution.

1. FLOWCHART

- i. **Determine if a year is a leap year:**



ii. Add two numbers and print the sum:



Print multiplication table for a number:

```

+-----+
|   Start   |
+-----+
|
| v
+-----+
| Input number |
+-----+
|
| v
  
```

+-----+

| For i = 1 to 10 |

| Print number * i |

+-----+

|

v

+-----+

| End |

+-----+

iii. Find HCF and LCM of two numbers:

+-----+

| Start |

+-----+

|

v

+-----+

|Input number1 and number2 |

+-----+

|

v

+-----+

| Calculate HCF and LCM |

+-----+

|

v

+-----+

| Print HCF and LCM |

+-----+

```

      |
      v
+-----+
| End      |
+-----+

```

iv. Keep taking inputs until 'x' is entered and print sum:

```

+-----+
| Start    |
+-----+

```

```

      |
      v
+-----+
| sum = 0   |
+-----+

```

```

      |
      v
+-----+
| Input number or 'x' |
+-----+

```

```

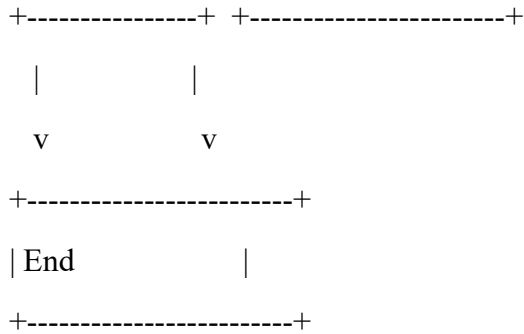
      |
      v
+-----+
| Is input = 'x'?     |
+-----+

```

```

      | Yes      | No
      v          v
+-----+ +-----+
| Print sum    | | sum = sum + input |

```



2. PSEUDOCODE

➤ Pseudocode for Leap Year Check

Input year

If (year % 4 == 0) Then

 If (year % 100 == 0) Then

 If (year % 400 == 0) Then

 Print "Leap Year"

 Else

 Print "Not a Leap Year"

 End If

 Else

 Print "Leap Year"

 End If

Else

 Print "Not a Leap Year"

End If

➤ Pseudocode for Sum of Two Numbers

Input number1, number2

sum = number1 + number2

Print sum

// Pseudocode for Multiplication Table

Input number

For i = 1 to 10

 Print number * i

End For

➤ **Pseudocode for HCF and LCM**

Input number1, number2

hcf = FindHCF(number1, number2)

lcm = (number1 * number2) / hcf

Print hcf, lcm

➤ **Pseudocode for Sum Until 'x' is Entered**

sum = 0

While true

 Input number

 If number == 'x' Then

 Break

 End If

 sum = sum + number

End While

Print sum

3. Using Java Codes in Android to implement the logic in my Program for this course.

My Java code:

```
import java.util.Scanner;
```

```
public class Main {
```

```
public static void main(String[] args) {  
    Scanner scanner = new Scanner(System.in);
```

a. Check if a year is a leap year

```
    System.out.println("Enter a year:");  
    int year = scanner.nextInt();  
    if (year % 4 == 0) {  
        if (year % 100 == 0) {  
            if (year % 400 == 0) {  
                System.out.println(year + " is a Leap Year");  
            } else {  
                System.out.println(year + " is not a Leap Year");  
            }  
        } else {  
            System.out.println(year + " is a Leap Year");  
        }  
    } else {  
        System.out.println(year + " is not a Leap Year");  
    }  
}
```

b. Add two numbers

```
    System.out.println("Enter two numbers:");  
    int number1 = scanner.nextInt();
```

c. `int number2 = scanner.nextInt();`

```
    int sum = number1 + number2;  
    System.out.println("Sum: " + sum);
```

d. Multiplication table

```

System.out.println("Enter a number to print its multiplication table:");
int num = scanner.nextInt();
for (int i = 1; i <= 10; i++) {
    System.out.println(num + " * " + i + " = " + (num * i));
}

```

e. Find HCF and LCM

```

System.out.println("Enter two numbers to find HCF and LCM:");
int num1 = scanner.nextInt();
int num2 = scanner.nextInt();
int hcf = findHCF(num1, num2);
int lcm = (num1 * num2) / hcf;
System.out.println("HCF: " + hcf + ", LCM: " + lcm);

```

f. Keep taking numbers until 'X' is entered

```

int totalSum = 0;
while (true) {
    System.out.println("Enter a number (or 'x' to stop):");
    String input = scanner.next();
    if (input.equals("x")) break;
    totalSum += Integer.parseInt(input);
}
System.out.println("Sum of all numbers: " + totalSum);
}

```

```

public static int findHCF(int a, int b) {
    while (a != b) {
        if (a > b) {

```



```
        a -= b;
    } else {
        b -= a;
    }
}
return a;
}
}
```

Q2. Using Android programming technology, Write a program asking the user to enter his/her marks (out of 100). Define a method that will display grades according to the marks entered below:

<i>Marks</i>	<i>Grade</i>
<i>91-100</i>	<i>AA</i>
<i>81-90</i>	<i>AB</i>
<i>71-80</i>	<i>BB</i>
<i>61-70</i>	<i>BC</i>
<i>51-60</i>	<i>CD</i>
<i>41-50</i>	<i>DD</i>
<i><=40</i>	<i>Fail</i>

The following question will be answered in four steps

- Create a New Android Project in Android Studio as my favorite IDE and my project will be called GradeCalculation with an empty activity,
- Update the XML Layout File,
- Update the Main Activity Code and finally,
- Run the Application

Step 1. Open Android Studio.

Create a new project with an empty activity and name I'll name my project "GradeCalculation" and I'll then choose the necessary settings that I want.

Step 2: Update the XML Layout File

In the activity_main.xml file, create the layout for the user to enter their marks and display the grade.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Enter your marks (out of 100):"
```

```
android:textSize="18sp" />
```

```
<EditText  
    android:id="@+id/editTextMarks"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:inputType="number" />
```

```
<Button  
    android:id="@+id/buttonSubmit"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Submit" />
```

```
<TextView  
    android:id="@+id/textViewGrade"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:textSize="18sp"  
    android:layout_marginTop="20dp" />
```

```
</LinearLayout>
```

Step 3: Update the Main Activity Code

In the MainActivity.java, we will write the logic to calculate and display the grade based on the entered marks. I'll be using Java as my favorite language for this program.

My Java Code will be:

```
package com.example.gradecalculator;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {
```

```

private EditText editTextMarks;

private TextView textViewGrade;

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);


    editTextMarks = findViewById(R.id.editTextMarks);
    textViewGrade = findViewById(R.id.textViewGrade);
    Button buttonSubmit = findViewById(R.id.buttonSubmit);
    buttonSubmit.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View v) {

            String marksStr = editTextMarks.getText().toString();

            if (!marksStr.isEmpty()) {

                int marks = Integer.parseInt(marksStr);

                String grade = calculateGrade(marks);

                textViewGrade.setText("Your grade is: " + grade);

            } else {

                textViewGrade.setText("Please enter your marks.");

            }

        }

    });

}

```

```

private String calculateGrade(int marks) {

    if (marks >= 91 && marks <= 100) {

        return "AA";

    } else if (marks >= 81 && marks <= 90) {

        return "AB";

    }

}

```

```
} else if (marks >= 71 && marks <= 80) {  
    return "BB";  
} else if (marks >= 61 && marks <= 70) {  
    return "BC";  
} else if (marks >= 51 && marks <= 60) {  
    return "CD";  
} else if (marks >= 41 && marks <= 50) {  
    return "DD";  
} else {  
    return "Fail";  
}  
}  
}
```

Step 4: Run the Application and enter a number to represent your marks

I then used my physical Android Phone to Build and run the application as the emulator is very heavy and consumes my disk space.

Enter the marks and press "Submit" to see the grade.

After the building and running process on my phone, I entered marks randomly and pressed the submit button to see my grades, and the results were based on the marks entered.