**Edinburgh College**

**NPA Software Development and Web**

**Software Design and Development**

**J27C76/008**

**Gregorian Calendar**

**Student Name: Adrián Sánchez Rodríguez**

**EC Number: EC1939656**

**Table of Content**

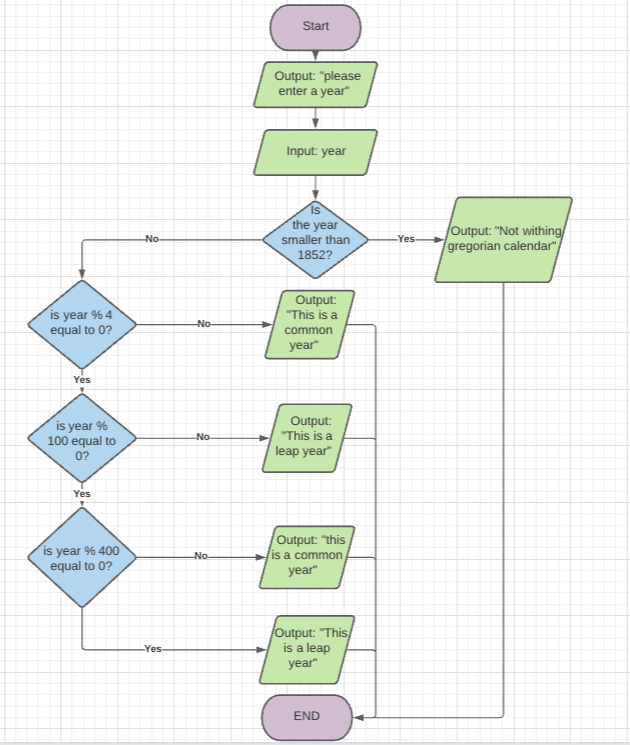
**1. Flowchart …………………………………………….……… pag.1**

**2. Pseudocode ………………………………………………… pag. 2**

**3. Source Code……………………………………………...… pag. 3**

**4. Test Scenario………………………………………….…… pag. 4**

**1- Flowchart**



**2- Pseudocode**

Write “Please type a year”;

Save the input and convert it to an integer;

If the year is smaller than 1852

Write “This year is not withing the Gregorian calendar;

For the rest of the numbers

If the remainder of the division of the year by 4 is not equal to 0

Write “this is a common year”

If the remainder of the division of the year by 100 is not equal to 0

Write “this is a leap year”

If the remainder of the division of the year by 400 is not equal to 0

Write “this is a common year”

For the rest of the numbers

Write “this is a leap year”

**3- Source Code**

/\*

Program: Gregorian Calendar Calculator

Description: A program to calculate wether a year is common or leap.

Author: Adrian Sanchez (EC1939656)

Company: Edinburgh College

Version: 1.0

\*/

using System;

using System.Runtime.CompilerServices;

namespace GregorianCalendar

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("\*\*\*Leap Year Calculator\*\*\*");

Console.WriteLine("Please type a year (Ex: 1748)");

string num = Console.ReadLine();

int year = Convert.ToInt32(num);

if (year < 1582)

{

Console.WriteLine("This year is not within the Gregorian Calendar");

}

else

{

if (year % 4 != 0)

{

Console.WriteLine("This is a common year");

}

else if (year % 100 != 0)

{

Console.WriteLine("This is a leap year");

}

else if (year % 400 != 0)

{

Console.WriteLine("This is a common year");

}

else

{

Console.WriteLine("This is a leap year");

}

}

}

}

}

**4- Test Scenario**

|  |  |
| --- | --- |
| **Test case 1 – Tester Name - Date** | |
| Test Data | Input “2000” |
| Expected behaviour | Output: “This is a leap year” |
| Actual behaviour | Output: “This is a leap year” |
| Comments / fixes | Worked as intended |

|  |  |
| --- | --- |
| **Test case 2 – Tester Name - Date** | |
| Test Data | Input “1999” |
| Expected behaviour | Output: “This is a common year” |
| Actual behaviour | Output: “This is a common year” |
| Comments / fixes | Working as intended |

|  |  |
| --- | --- |
| **Test case 3 – Tester Name - Date** | |
| Test Data | Input “1748” |
| Expected behaviour | Output: “This is a leap year” |
| Actual behaviour | Output: “This is a leap year” |
| Comments / fixes | Working as intended |

|  |  |
| --- | --- |
| **Test case 4 – Tester Name - Date** | |
| Test Data | Input: “1436” |
| Expected behaviour | Output: “This year is not within the Gregorian Calendar” |
| Actual behaviour | Output: “This year is not within the Gregorian Calendar” |
| Comments / fixes | Working as intended |

|  |  |
| --- | --- |
| **Test case 5 – Tester Name - Date** | |
| Test Data | Input: “shdjskdajkdh” |
| Expected behaviour | No output. Program will crash |
| Actual behaviour | Program crashes |
| Comments / fixes | This is because the program has no way to validate whether the input is a number or not, and tries to convert it to an integer, resulting in a crash. Further investigation on how to identify the input as a string and add an option into the loop to let the user know that their input is invalid. |