# **ASIAN COLLEGE OF HIGHER STUDIES**

**(Affiliated to Tribhuvan University)**

**Dhobidhara, Kathmandu**



**REPORT ON  
“CALENDAR PROJECT”**

**“OOP”**

**(CSC-161)**

**Submitted by**

**Ojashwi Shrestha (191807)**

**Ashmin Kumar Khadka (191839)**

**“Second Semester”**

**Submitted to**

**Mr. Prarup Gurung**

**ACHS, Kathmandu**

**April , 2021**

**LETTER OF APPROVAL**

This is to certify that this project prepared by Ojashwi Shrestha and Ashmin Kumar Khadka entitled “Calendar Project” in the partial fulfillment of the requirement for the degree of BSc. Computer Science and Information Technology has been well studied. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

Signature:

Internal Examiner:

Prarup Gurung

Lecturer

Asian College of Higher Studies

**ACKNOWLEDGEMENT**

This is the OOP project of Ojashwi Shrestha and Ashmin Kumar Khadka of Bsc.CSIT second semester. This project is a simple catalogue of a calendar. We have been able to learn a lot about the Object Oriented Programming language with the completion of this assignment. So, we would like to express a thanks to our teacher Mr. Prarup Gurung for providing their invaluable guidance, comments, and suggestions throughout the course of the project. We would especially like to thank to all the teachers for constantly motivating to work harder.

**ABSTRACT**

This is a simple catalogue of a calendar. In the program, the user has two choices to view the date. The first option requires the user to enter a valid day, month and year. After processing the output will be shown of the respective day the entered date had been or will be. The second option requires the user to enter a desired month and year. Then the calendar layout of the entered month of that year will be given as output. This current build of the project supports from the year 1900 to 2400. There is also a third option which exits the current running program.

**TABLE OF CONTENTS**

[CHAPTER 1: INTRODUCTION 1](#_Toc69065627)

[1.1 Introduction to Project 1](#_Toc69065628)

[1.2 Software Development Life Cycle 1](#_Toc69065629)

[ System study 1](#_Toc69065630)

[ System Analysis 1](#_Toc69065631)

[ Coding 2](#_Toc69065632)

[ Development 2](#_Toc69065633)

[ Checking 2](#_Toc69065634)

[ Implementation 2](#_Toc69065635)

[ Maintenance 2](#_Toc69065636)

[CHAPTER 2: REQUIREMENTS 4](#_Toc69065637)

[2.1 List of Features 4](#_Toc69065638)

[CHAPTER 3: IMPLEMENTATION 5](#_Toc69065639)

[Source Code 5](#_Toc69065640)

[CHAPTER 4: TESTING 6](#_Toc69065641)

[Screenshots 6](#_Toc69065642)

[CONCLUSION 7](#_Toc69065643)

# **CHAPTER 1: INTRODUCTION**

## **1.1 Introduction to Project**

A quiz is aform of game or mind sport, in which the players attempt to answer questions correctly. In this project, we have a game to test the knowledge about a certain subject. An individual will be choosing one category among three and answer the questions related to the chosen category. We have used branching statements for this project.

## **1.2 Software Development Life Cycle**

The software development life cycle (SDLC) is a framework defining tasks performed at each step in the software development process. SDLC is a structure followed by a development team within the software organization. It consists of a detailed plan describing how to develop, maintain and replace specific software. The life cycle defines a methodology for improving the quality of software and the overall development process. The software development life cycle is also known as the software development process.

The stages for the software development life cycle are as follows:

## **System study**

In order to develop a software for any system, the system should be studied. This helps to develop the software on the basis of peripherals in the system.

## **System Analysis**

After studying the system, the system needs to be analyzed properly. For a smooth development of a software, this stage provides information about the system and allows to develop desirable software.

## **Coding**

After studying and analyzing the system, the activity of putting the software in written form is proceeded. This activity is called coding. Algorithms and flowcharts can also be made for the software.

## **Development**

After the coding process, a software is developed on the basis of system study and system analysis.

* **Checking**After the development of a software, it needs to be checked whether it is desirable or not. This stage is responsible for this and proceeded to implementation.

## **Implementation**

After checking, if the software is confirmed to be desirable, it is implemented. It is free from loops and other defects. It is launched and is ready to be used by the users.

## **Maintenance**

When the software is launched, it is maintained often. If any problems or loop is found in the software, it is fixed by letting the users know about it through maintenance break. Maintenance break is done if there are any updates also.

1. System Study

7) Maintenance

2) System Analysis

3) Coding

6) Implementation

5) Checking

4) Development

**Figure 1: System Development Life Cycle**

# **CHAPTER 2: REQUIREMENTS**

## **2.1 List of Features**

The key features of the calendar are as follows:

1. There are two options for the user to choose from. These options are the different ways to view the calendar.
2. Option 1:

Enter the date, month and year to view the exact day of that date ie. Sunday, Monday, Tuesday, etc.

1. Option 2:

Enter the month and year to view the full calendar of that particular month in that year.

# **CHAPTER 3: IMPLEMENTATION**

## **Source Code:**

#include<iostream>

#include<conio.h>

#include<ctime>

#include<Windows.h>

using namespace std;

void heading(char ch, int i)

{

int j;

for (j = 0; j <= i; j++)

cout << ch;

}

class calendar

{

private:

int date, month,year;

public:

calendar(int d = 1, int m = 1, int y = 1900)

{

date = d;

month = m;

year = y;

}

void day(int);

void monthcal(int,int);

bool leapyear(int);

int DaysInMonth(int,int);

int DayNumber(int,int,int);

};

bool calendar :: leapyear(int y)

{

bool Leap;

if ((y % 4 == 0) && ((y % 100 != 0) || (y % 400 == 0)))

Leap = true;

else

Leap = false;

return Leap;

}

int calendar :: DaysInMonth(int m, int y)

{

switch (m)

{

case 1:

case 3:

case 5:

case 7:

case 8:

case 10:

case 12: return 31;

case 2: if (leapyear(y))

return 29;

else

return 28;

default: return 30;

}

}

int calendar :: DayNumber(int d, int m, int y)

{

int i;

int dow = 6;

for (i = 1583; i<y; i++)

{

dow += (leapyear(i)) ? 2 : 1;

dow+=364;

}

for (i = 1; i<m; i++)

dow += DaysInMonth(i, y);

dow += d;

return dow;

}

void calendar::day(int a)

{

switch (a)

{

case 1:

cout << "MONDAY";

break;

case 2:

cout << "TUESDAY";

break;

case 3:

cout << "WEDNESDAY";

break;

case 4:

cout << "THURSDAY";

break;

case 5:

cout << "FRIDAY";

break;

case 6:

cout << "SATURDAY";

break;

case 0:

cout << "SUNDAY";

break;

default:

cout << "\a\a\a\n\nERROR!!!";

}

}

void calendar::monthcal(int month, int year)

{

int j, mon, c;

j = 0;

for (j = 0; j <= 6; j++) // For printing 7 days of week

{

day(j);

cout << " ";

}

j = 1;

c = DayNumber(--j, month, year);

c=c%7;

cout << "\n";

switch (c)

{

case 0:

break;

case 1:

cout << "\t";

break;

case 2:

cout << "\t\t";

break;

case 3:

cout << "\t\t\t";

break;

case 4:

cout << "\t\t\t\t";

break;

case 5:

cout << "\t\t\t\t\t";

break;

case 6:

cout << "\t\t\t\t\t\t";

}

mon = DaysInMonth(month, year);

for (j = 1; j <= mon; j++) //Printing the month's calendar

{

cout << " " << j << "\t";

c++;

if (!(c % 7))

cout << "\n";

}

}

int main()

{

int x,y,z;

char ch; char t = '\xCD';

do

{

system("cls");

int i, date, year, month;

cout << "\t\t CALENDAR";

cout << "\n\t\t -----------------------------------------";

cout << "\n\t\t Press 1: The day of a specific date.\n\t\t Press 2: View the calendar of any given month.\n\t\t Press 3: Exit. ";

cout << "\n\t\t -----------------------------------------";

cout << endl << endl << "\n\t\t Press your choice: ";

cin >> i;

if (i == 1) //Option 1

{

cout << "\n\t\t Enter date(dd):";

cin >> date;

if (date>31 || date<1)

{

cout << "\n\t\t Please provide valid date.\n\t\t Press any key";

getch();

system("cls");

main();

}

month:

cout << "\n\t\t Enter month(mm):";

cin >> month;

if (month>12 || month<1)

{

cout << "\n\t\t Please provide valid month.\n\t\t Press any key";

getch();

goto month;

}

year:

cout << "\n\t\t Enter year(yyyy):";

cin >> year;

if (year>2400 || year<1900)

{

getch();

goto year;

}

calendar obj(date, month, year);

i = obj.DayNumber(--date, month, year);

cout << "\n\t\t" << ++date << "/" << month << "/" << year << " is ";

i=i%7;

obj.day(i);

}

else if (i == 2) //Option 2

{

cout << "\n\t\t Enter month(mm):";

cin >> month;

month2:

if (month>12 || month<1)

{

cout << "\n\t\t Please provide valid month.\n\t\t Press any key";

getch();

goto month2;

}

cout << "\n\t\t Enter year(yyyy):";

cin >> year;

year2:

if (year>2400 || year<1900)

{

getch();

goto year2;

}

calendar obj;

obj.monthcal(month, year);

}

else if (i == 3)

{

exit(0);

}

cout << "\n\t\t Would you like to try again (y/n):";

ch = \_getch();

}

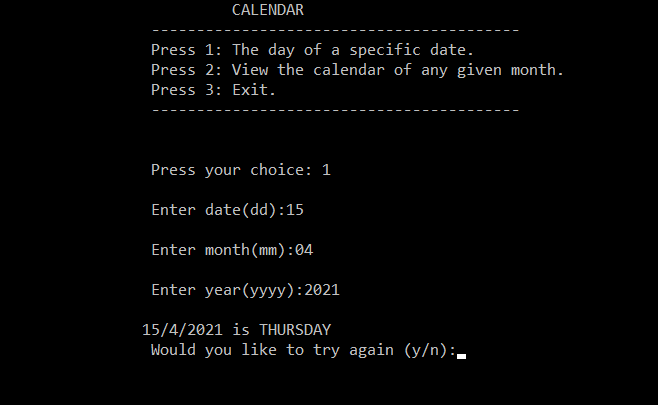
while (ch == 'y' || ch == 'Y');

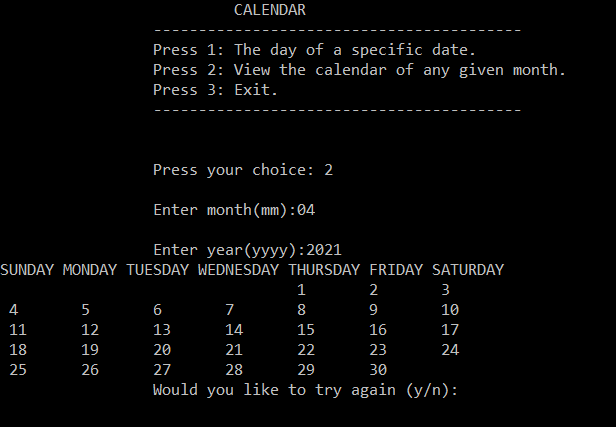
return 0;

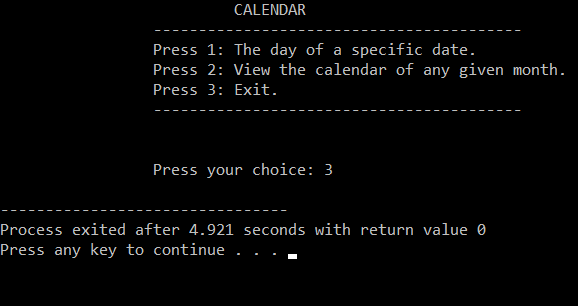
}

# **CHAPTER 4: TESTING**

## **Screenshots:**







# **CONCLUSION**

We would like to express our sincere gratitude to our teacher Mr. Prarup Gurung for providing us with this golden opportunity to explore more about the subject through this assignment. This project has enabled us to put all the ideas we have grasped to a good use and create something we are proud to present. We really hope this project reaches up to the required standards.