PROJECT REPORT ON CALENDAR APPLICATION IN C

SUBMITTED BY :

- SARTHAK RANA - 2022A6R044
- AYUSHMAN VOHRA - 2022A6R059
- NAMYAA SARIN - 2022A6R050
- SARTHAK SAHOO - 2022A6R039

CLASS : CSE AI/ML

SECTION : A1

SUBJECT : C PROGRAMMING (LAB)

DATE OF SUMBISSION :

SUBMITTED TO

- MR. SUNIL KUMAR
- MRS. MEENU

Table of contents:

Problem statement:

Making a calendar of an year in c language

Objectives:

- Planning our daily activities.
- Keeping a track of events.
- Staying organized and enhancing productivity.
- Managing a daily schedule.
- Remembering important dates.
- Alleviating anxiety and stress.
- Knowing the important festival dates.
- Remembering birthdays and keeping commitments.
- useful exposure to the C Programming language.
- work effectively as a group and manage all the tasks effectively.
- learning & enhance our ability in C Programming.

Introduction:

"This report has described the successful design and development of a calendar program. This report outlines the design and development of a computer software system to code blocks. The program was written in C language. Basically three operations can be done in this calendar application. To find out the day corresponding to a given date, the date, month and year are asked. You can list the days and dates of any month of any year.

Abstract:

It’s a calendar program which will prints a calendar of our whole year. The purpose if this program is to basically learning of our C language in programming. The calendar program application presented here is a very simple console application developed using C programming language. It is built without using any graphic properties: instead it utilizes many windows properties to give application a colourful look and feels. It is compiled in code::blocks using GCC compiler. Background

KNOWLEDGE

As a group our first exposure to C programming was the simple yet famous “hello world!” program. The following construct illustrates this simple program and shows some of the syntax used.

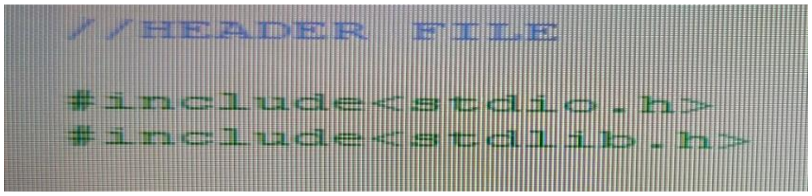
As a language, C is relatively easy to learn though it is more cryptic and example above is broken down into the following steps: the #include is a pre-processor directive. The pre-processor is the first tool to read the source code and it is instructed to substitute the entire line with the <stdio.h>.The next line indicates that a function named “main” is being defined. The “main” function serves a special purpose in C programming; it is used to begin program execution. The “int” is a type specifier and its purpose is to indicate that the value returned to the invoker as a result of reading the main function, is an integer. The keyword “void” as a parameter list indicates that the main function takes no arguments. #include <stdio.h>int main(void){ printf("hello world\n");return 0;}

Body of project:

Libraries:

This project includes two libraries:

1) #include<stdio.h
2) #include<stdlib.h>



```
//HEADER FILE

#include<stdio.h>
#include<stdlib.h>
```

#include<stdio.h>:

The C programming language provides many standard library functions for file input and output. These functions make up the bulk of the C standard library header <stdio.h>.

#include<stdio.h> is a statement which tells the compiler to insert the contents of stdio at that particular place.□The first thing you will notice is the first line of the file, the #include "stdio.h" line. This is very much like the #define the preprocessor , except that instead of a simple substitution, an entire file is read in at this point.□The system will find the file named "stdio.h" and read its entire contents in, replacing this statementstdio.h is the header file for standard input and output. This is useful for getting the input from the user(Keyboard) and output result text to the monitor(screen). With out this header file, one can not display the results to the users on the screen or cannot input the values through the keyboard.

#include<stdlib.h>:

The "stdlib.h" is a header file(library). Its full form is Standard Library. It has many functions(piece of code dedicated for performing specific task) written in it. Some of them are:

- malloc()
- calloc()
- free()
- exit()

The <stdlib.h> contains essential functions and data variable declarations for memory allocation and revocation. In other words, it allows you to use functions such as malloc() and calloc() to allocate memory to a particular objects and eventually revoke it by free()

- Body of project:

We have used these main things:

1) Do statement
2) For loop
3) Function

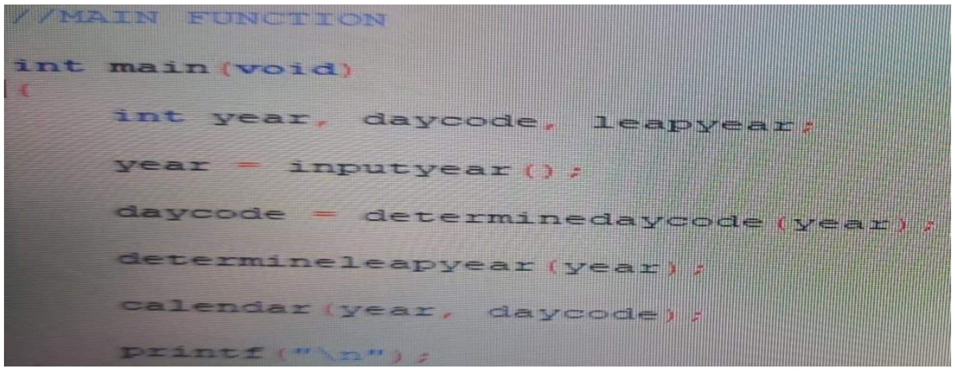
Function:

A function is a group of statements that together perform a task. Every C program has at least one function, which is main(), and all the most trivial programs can define additional functions. A function declaration tells the compiler about a function's name, return type, and parameters. A function definition provides the actual body of the function.

Main function:

In C, the "main" function is treated the same as every function, it has a return type (and in some cases accepts inputs via parameters). The only difference is that the main function is "called" by the operating system when the user runs the program. Thus the main function is always the first code executed when a program starts.

When main calls a function, it passes the execution control to that function. The function returns control to main when a return statement is executed or when end of function is reached. In our main function firsly we declare our variables which we have to use in our function.



```
//MAIN FUNCTION
int main(void)
{
    int year, daycode, leapyear;
    year = inputyear();
    daycode = determinedaycode(year);
    determineleapyear(year);
    calendar(year, daycode);
    printf("\n");
```

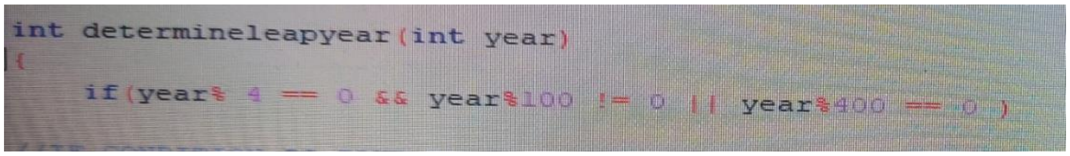
Variables:

1) Year
2) Day
3) Day in month
4) Weekday
5) First day
6) Month

Data type of all these variables is int. but also another variable which have a data type of character. Char ch

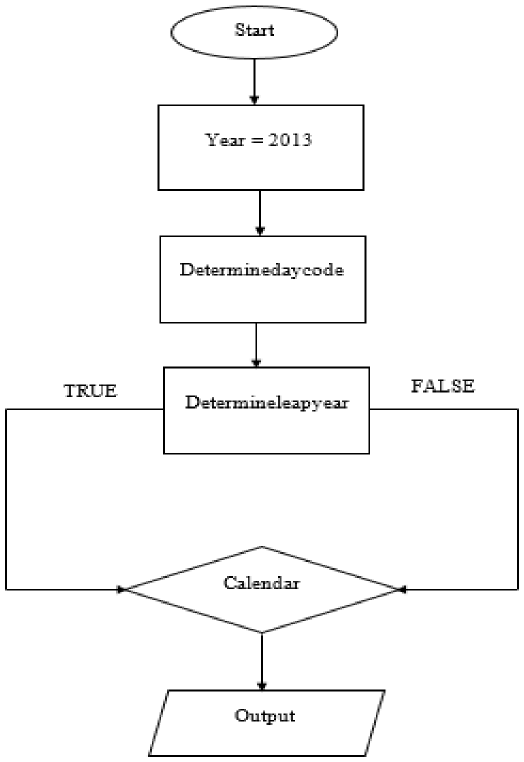
IF statement:

The if statement is the ability to control the flow of your program, letting it make decisions on what code to execute, is valuable to the programmer. The if statement allows you to control if a program enters a section of code or not based on whether a given condition is true or false.

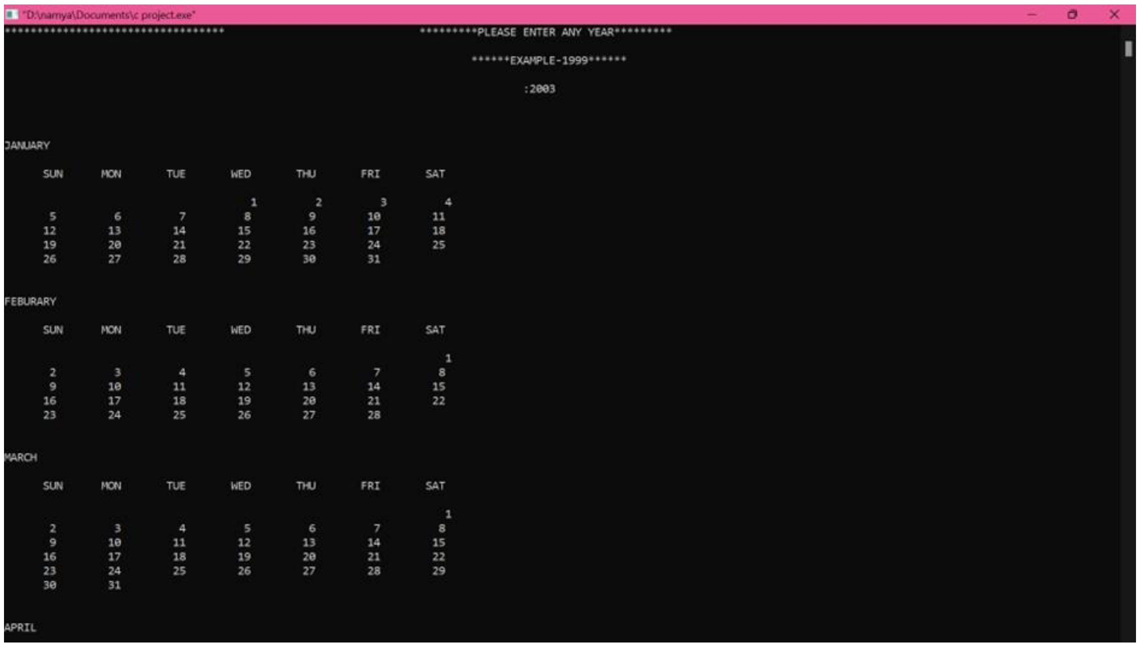


```
int determineleapyear(int year)
{
    if(year% 4 == 0 && year%100 != 0 || year%400 == 0 )
```

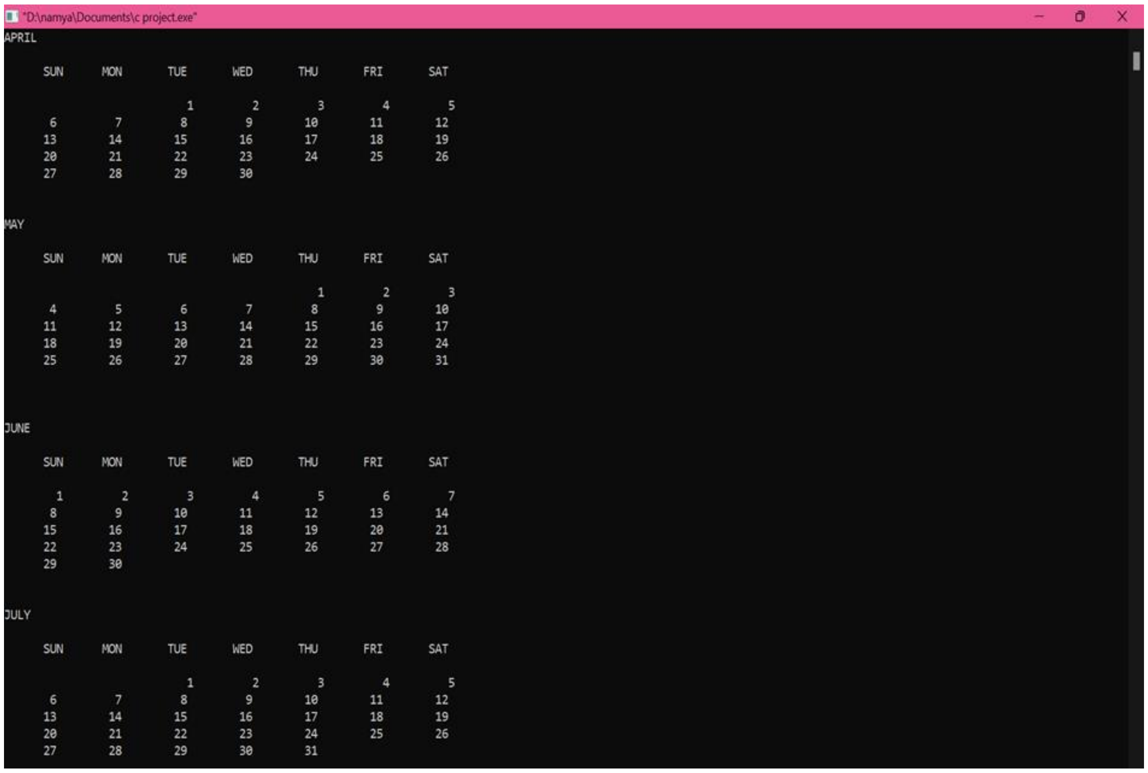
FLOWCHART:



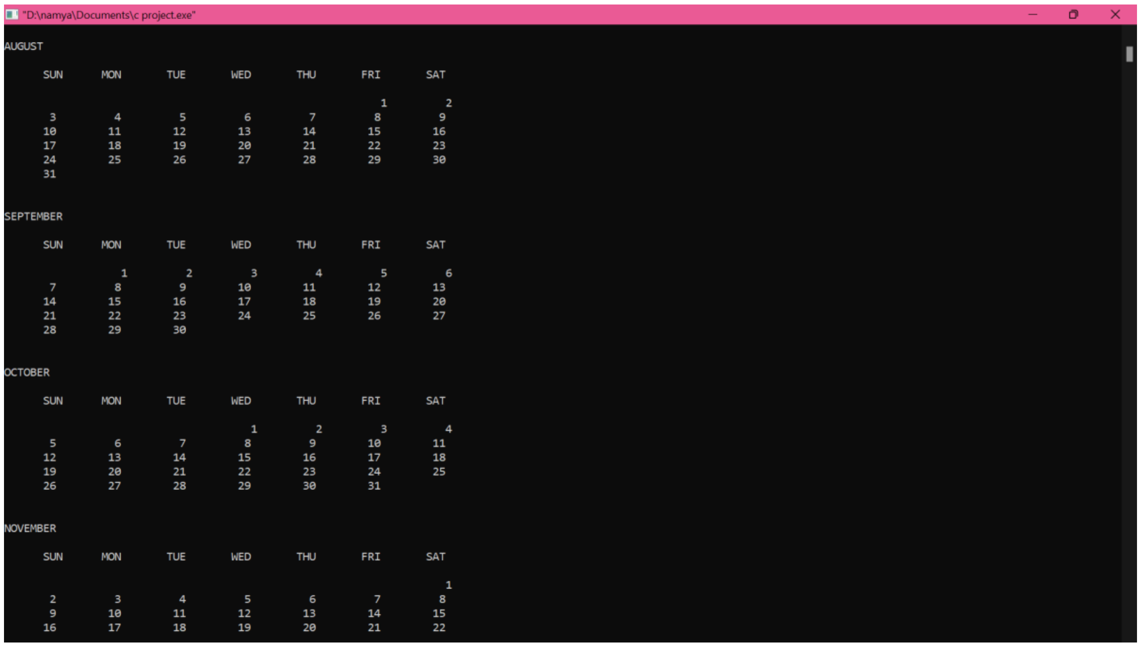
- Output:



```
"D:\namya\Documents\c project.exe"
**********************************                              *********PLEASE ENTER ANY YEAR*********

                                                                        ******EXAMPLE-1999******

                                                                                :2003


JANUARY

      SUN     MON     TUE     WED     THU     FRI     SAT

                                1       2       3       4
        5       6       7       8       9      10      11
       12      13      14      15      16      17      18
       19      20      21      22      23      24      25
       26      27      28      29      30      31


FEBURARY

      SUN     MON     TUE     WED     THU     FRI     SAT

                                                        1
        2       3       4       5       6       7       8
        9      10      11      12      13      14      15
       16      17      18      19      20      21      22
       23      24      25      26      27      28


MARCH

      SUN     MON     TUE     WED     THU     FRI     SAT

                                                        1
        2       3       4       5       6       7       8
        9      10      11      12      13      14      15
       16      17      18      19      20      21      22
       23      24      25      26      27      28      29
       30      31


APRIL
```

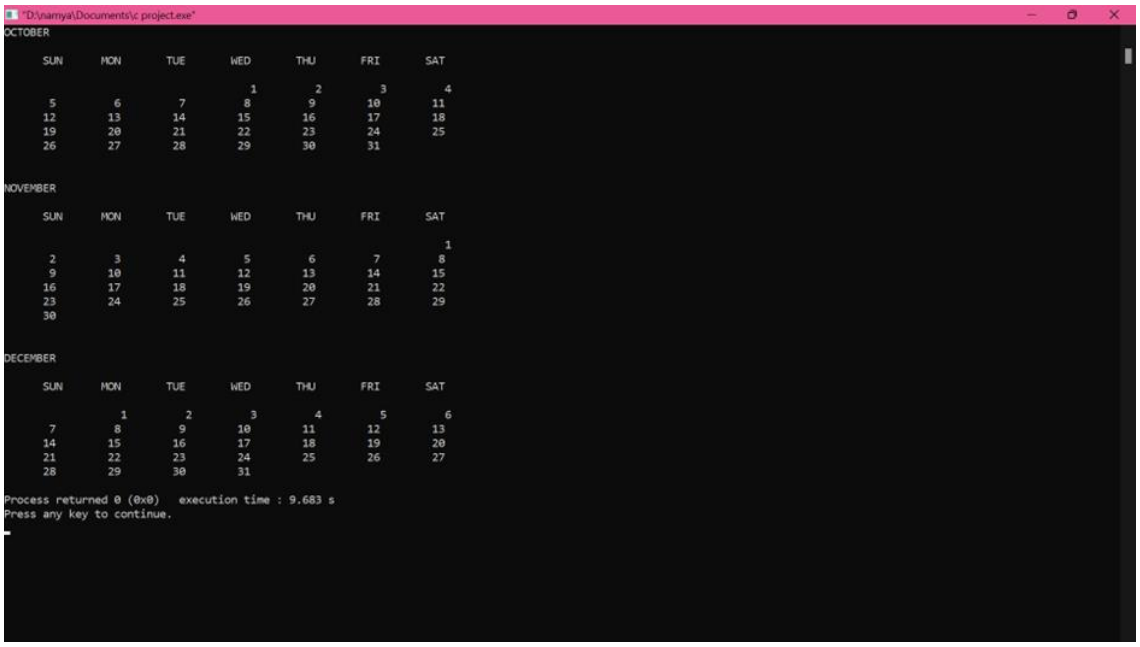


```
"D:\namya\Documents\c project.exe"
APRIL

      SUN     MON     TUE     WED     THU     FRI     SAT

                        1       2       3       4       5
        6       7       8       9      10      11      12
       13      14      15      16      17      18      19
       20      21      22      23      24      25      26
       27      28      29      30


MAY

      SUN     MON     TUE     WED     THU     FRI     SAT

                                        1       2       3
        4       5       6       7       8       9      10
       11      12      13      14      15      16      17
       18      19      20      21      22      23      24
       25      26      27      28      29      30      31



JUNE

      SUN     MON     TUE     WED     THU     FRI     SAT

        1       2       3       4       5       6       7
        8       9      10      11      12      13      14
       15      16      17      18      19      20      21
       22      23      24      25      26      27      28
       29      30


JULY

      SUN     MON     TUE     WED     THU     FRI     SAT

                        1       2       3       4       5
        6       7       8       9      10      11      12
       13      14      15      16      17      18      19
       20      21      22      23      24      25      26
       27      28      29      30      31
```



"D:\namya\Documents\c project.exe"

```
AUGUST

     SUN     MON     TUE     WED     THU     FRI     SAT

                                               1       2
      3       4       5       6       7       8       9
     10      11      12      13      14      15      16
     17      18      19      20      21      22      23
     24      25      26      27      28      29      30
     31


SEPTEMBER

     SUN     MON     TUE     WED     THU     FRI     SAT

               1       2       3       4       5       6
      7       8       9      10      11      12      13
     14      15      16      17      18      19      20
     21      22      23      24      25      26      27
     28      29      30


OCTOBER

     SUN     MON     TUE     WED     THU     FRI     SAT

                               1       2       3       4
      5       6       7       8       9      10      11
     12      13      14      15      16      17      18
     19      20      21      22      23      24      25
     26      27      28      29      30      31


NOVEMBER

     SUN     MON     TUE     WED     THU     FRI     SAT

                                                       1
      2       3       4       5       6       7       8
      9      10      11      12      13      14      15
     16      17      18      19      20      21      22
```



"D:\namya\Documents\c project.exe"

```
OCTOBER

     SUN     MON     TUE     WED     THU     FRI     SAT

                               1       2       3       4
      5       6       7       8       9      10      11
     12      13      14      15      16      17      18
     19      20      21      22      23      24      25
     26      27      28      29      30      31


NOVEMBER

     SUN     MON     TUE     WED     THU     FRI     SAT

                                                       1
      2       3       4       5       6       7       8
      9      10      11      12      13      14      15
     16      17      18      19      20      21      22
     23      24      25      26      27      28      29
     30


DECEMBER

     SUN     MON     TUE     WED     THU     FRI     SAT

               1       2       3       4       5       6
      7       8       9      10      11      12      13
     14      15      16      17      18      19      20
     21      22      23      24      25      26      27
     28      29      30      31

Process returned 0 (0x0)   execution time : 9.683 s
Press any key to continue.
_
```

❖ Source Code :

```
/*PROGRAM TO PRINT CALENDAR OF ANY YEAR
YOU WOULD LIKE TO SEE*/

//HEADER FILE

#include<stdio.h>
#include<stdlib.h>

//INITIALIZE THE ARRAYS

int
days_in_month[]={0,31,28,31,30,31,30,31,31,30,31,30,31};

char *months[]=

{
        " ",
        "\n\n\nJANUARY",
        "\n\n\nFEBURARY",
        "\n\n\nMARCH",
        "\n\n\nAPRIL",
        "\n\n\nMAY",
        "\n\n\nJUNE",
        "\n\n\nJULY",
        "\n\n\nAUGUST",
        "\n\n\nSEPTEMBER",
        "\n\n\nOCTOBER",
        "\n\n\nNOVEMBER",
        "\n\n\nDECEMBER"
};

//FUNCTION IS USED TO TAKE ANY YEAR AS INPUT
FROM THE USER

//EXAMPLE : 2000

int inputyear(void)
{
        int year;

        printf("\t\t\t\t\t\t\t\t*********WELCOME TO MIET
CALENDAR*********\n\n");

        printf("\n\n\n\n\n\n\n\n\n\n\n\tCREATED
BY\n***************************************\n\tSARTHA
K RANA\n\tAYUSHMAN\n\tNAMYAA
SARIN\n\tSARTHAK
SAHOO\n*********************************");

        printf("\t\t\t\t*********PLEASE ENTER ANY
YEAR*********\n\n\t\t\t\t\t\t\t\t******EXAMPLE-
1999******\n\n\t\t\t\t\t\t\t\t\t:");

        scanf("%d",&year);

        return year;

}

//CREATE A FUNCTION NAMED DETERMINE DAY
CODE

//FUNCTION IS USED TO INDIVIDUALLY KNOW THE
DAYS OF THE MONTH

int determinedaycode(int year)
{
        int daycode;
```

```
    int d1, d2, d3;

    d1 = (year - 1.)/ 4.0;
    d2 = (year - 1.)/ 100.;
    d3 = (year - 1.)/ 400.;

    daycode = (year + d1 - d2 + d3) %7;

    return daycode;
}

//CREATE A FUNCTION NAMED DETERMINE LEAP
YEAR

//THIS FUNCYION IS USED TO FIND IF THE YEAR
INPUT BY USER IS LEAP YEAR OR NOT

int determineleapyear(int year)
{
    if(year% 4 == 0 && year%100 != 0 || year%400 == 0 )

//IF CONDITION IS TRUE THAN THE YEAR IS A LEAP
YEAR HENCE FEBURARY WILL BE OF 29 DAYS
```

```
        {
                days_in_month[2] = 29;

        }

//IF CONDITION IS FALSE FEBURARY WILL BE OF 28
DAYS

        else

        {
                days_in_month[2] = 28;

        }
}

// CALLING VALUE OF YEAR AND DAYCODE IN
CALENDAR FUNCTION

//FUNCTION WILL PRINT THE NUMBER OF MONTHS
AND NUMBER OF DAYS ACCORDINGLY

void calendar(int year, int daycode)
{
        int month, day;

        for ( month = 1; month <= 12; month++ )

        {
                printf("%s", months[month]);

                printf("\n\n      SUN      MON       TUE       WED
THU       FRI       SAT\n\n" );

// CORRECT THE POSITION OF THE DAY

                for ( day = 1; day <= 1 + daycode * 5; day++ )

                {
                        printf("  ");
                }

// PRINT ALL THE DAYS IN MONTHS

                for ( day = 1; day <= days_in_month[month];
day++ )
```

```
            {
                printf("%7d",day);

// IS DAY BEFORE SATURDAY? ELSE START NEXT
LINE SUNDAY

                if ( ( day + daycode ) % 7 > 0 )
            {

                    printf("   " );

            }
                else
            {

                    printf("\n " );

            }
            }
                // SET POSITION FOR NEXT MONTH

                daycode = ( daycode + days_in_month[month]
) % 7;
```

```
        }
}

//MAIN FUNCTION

int main(void)
{
        int year, daycode, leapyear;

        year = inputyear();

        daycode = determinedaycode(year);

        determineleapyear(year);

        calendar(year, daycode);

        printf("\n");
}
```

References:

- All classes conducted .
- All c programming labs conducted
- Google search .