**Part A: Multiple Choice: Select ONE Correct Answer (14 Marks, Put the answers in the table or no marks!!!)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **D** | **D** | **A** | **D** | **B** | **D** | **A** |

1 To assign a value 1 to variable x, you write

A. x := 1;

B. 1 := x;

C. 1 = x;

D. x = 1;

E. x == 1;

2 The ASCII of 'a' is 97. What is the ASCII for 'c'?

A. 97

B. 98

C. 96

D. 99

3 Which of the following statement prints smith\exam1\test.txt?

A. cout << "smith\\exam1\\test.txt";

B. cout << "smith"\exam1"\test.txt";

C. cout << "smith\"exam1\"test.txt";

D. cout << "smith\exam1\test.txt";

4 A character is stored in \_\_\_\_\_\_\_\_\_\_.

A. three bytes

B. two bytes

C. four bytes

D. one byte

5 Suppose x is 1. What is x after x -= 1?

A. -1

B. 0

C. 1

D. 2

E. -2

6 The extension name of a C++ source code file is

A. .class

B. .java

C. .exe

D. .cpp

E. .obj

7 To improve readability and maintainability, you should declare \_\_\_\_\_\_\_\_\_ instead of using literal values such as 3.14159.

A. constants

B. variables

C. classes

D. functions

**Part B: True or False (6 Marks, Put the answers T/F in the table or no marks!!!)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** |
| **T** | **F** | **F** | **F** | **T** | **F** |

1. The CPU understands instructions written in a binary machine language.
2. Comments in C/C++ begin only with the # character.
3. In C/C++, math expressions are always evaluated from left to right, no matter what the operators are.
4. C/C++ allows you to compare strings, but it cannot be case sensitive.
5. Decision structures are also known as selection structures.
6. When a piece of data is read from a file, it is copied from the file into the program

**Part C Answer the Questions. (15 Marks)**

Complete the program as requested below in C/C++. Attach the flowchart, source codes (text format, not image so I can run the program) and also the screen shots of input and output in this document.

***Find the number of days in a month***

Write a program that prompts the user to enter the month and year and displays the number of days in the month.

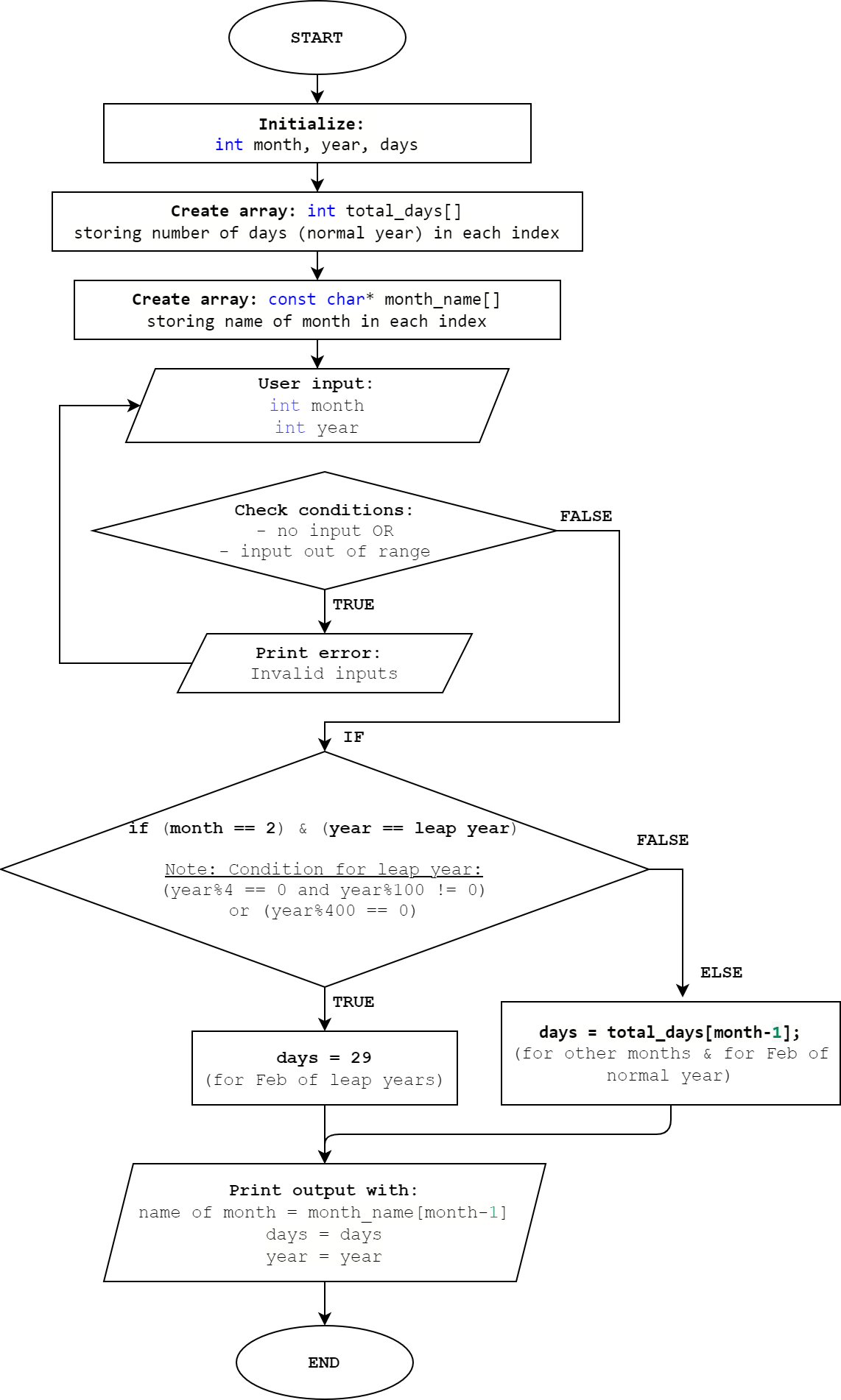
For example, if the user entered month 2 and year 2012, the program should display

February 2012 has 29 days

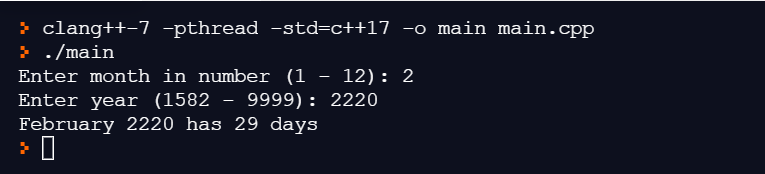
If the user entered month 3 and year 2015, the program should display

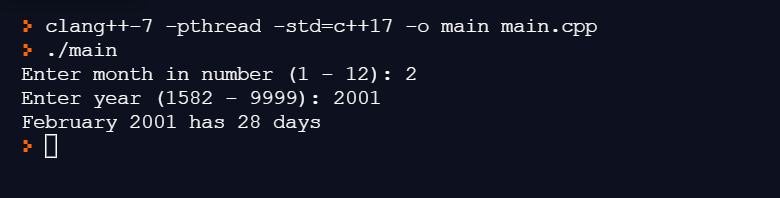
March 2015 has 31 days

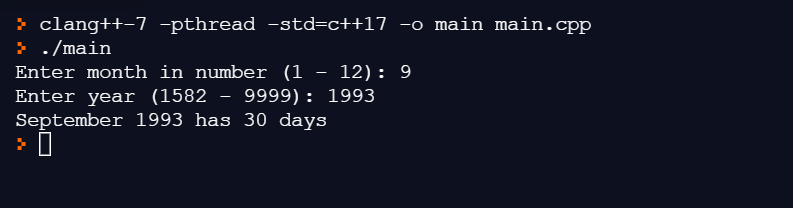
1. **Flowchart:**



1. **Source codes:**
2. /\*C++ program to find number of days in a month\*/
4. #include <iostream>
5. #include<ios> //to get stream size
6. #include<limits> //to get numeric limits
8. **using** **namespace** std;
10. **int** main(**void**) {
11. **int** month = 0, year = 0, days = 0;
13. /\* Use integer array & array of pointers to save
14. total days (normal year) & name of the months in monthly order \*/
15. **int** total\_days[] = {31,28,31,30,31,30,31,31,30,31,30,31};
16. **const** **char**\* month\_name[] = {
17. "January", "February", "March", "April",
18. "May", "June", "July", "August", "September",
19. "October", "November", "December"
20. };
22. /\* Get inputs (month & year) from user
23. Run loop if input fails (no input, input out of range)\*/
24. **while** ((cout << "Enter month in number (1 - 12): ")
25. && (!(cin >> month) || (month < 1) || (month > 12)))
26. {
27. cin.clear(); // clear bad input flag
28. cin.ignore(numeric\_limits<streamsize>::max(), '\n'); //Ignore input
29. cout << "Invalid input. Try again.\n";
30. }
32. **while** ((cout << "Enter year (1582 - 9999): ")
33. && (!(cin >> year) || (year < 1582) || (year > 9999)))
34. {
35. cin.clear();
36. cin.ignore(numeric\_limits<streamsize>::max(), '\n');
37. cout << "Invalid input. Note: Leap years started from 1582.\n";
38. }
40. /\* Total days is 29 if user input Feb & leap year
41. Leap year: divisible by 4, but not by 100. Unless divisible by 400 \*/
42. **if** ((month == 2) && ((year%4 == 0 && year%100 != 0)||(year%400 == 0)))
43. {
44. days = 29;
45. }
46. /\* Else: Get total days of month from saved array (index = month - 1)\*/
47. **else**
48. {
49. days = total\_days[month-1];
50. }
52. /\* Print out result
53. Get name of month from saved array (index = month - 1) \*/
54. cout << month\_name[month-1] << " " << year << " " << "has "
55. << days << " "  << "days\n";
57. **return** 0;
58. }
59. **Screenshots of inputs & outputs:**

**Example 1:** Leap year: Feb 2220

**Example 2:** Normal year: Feb 2001

**Example 3:** Another month: Sep 1993

**Example 4:** Handle wrong inputs