

```

1  #include<stdio.h>
2  #include<conio.h>
3
4  typedef struct date
5  {
6      int day,month,year;
7  } Date;
8
9  void display(Date date)
10 {
11     printf("%d / %d / %d\n",date.day,date.month,date.year);
12 }
13
14 int main()
15 {
16     Date input,result;
17
18     //dpm = day per each month, used to store no.of days associated with given month
19     int dpm;
20     /*
21     an array that stores no.of days associated with every month,
22     for example january has 31 days so day_per_month[0] will give 31
23     similarly if the user gives the month as 12 then the day will be
24     day_per_month[givenMonth-1] = 31 days
25     */
26     int day_per_month[] = {31,28,31,30,31,30,31,31,30,31,30,31};
27
28     printf("\n***Enter the date***\nYear:");
29     scanf("%d",&input.year);
30
31     /*
32     Iteratively forcing the user to give a valid year.
33     Keynotes: year will never be negative, nor an astronomical number
34     */
35     while(input.year<0 || input.year>99999)
36     {
37         printf("Please enter a valid Year!!\nYear:");
38         scanf("%d",&input.year);
39     }
40     printf("Month:");
41     scanf("%d",&input.month);
42
43     /*
44     Iteratively forcing the user to give a valid month.
45     Keynotes: month never be less than 1 or greater than 12
46     */
47     while(input.month>12 || input.month<1)
48     {
49         printf("Please Enter a valid Month!!\nMonth:");
50         scanf("%d",&input.month);
51     }
52     /*
53     assigning no.of days into 'dpm' according to given month by the user,
54     to verify : is the user gives a valid day or not
55     (no.of days varies per month)
56     */
57     dpm = day_per_month[input.month-1];
58
59     //if given month is february then checking for leap year
60     if (input.month == 2)
61         if (((input.year % 4 == 0) && (input.year % 100 != 0)) || (input.year % 400 == 0)) dpm=29;

```

```

61     if (((input.year % 4 == 0) && (input.year % 100 != 0)) || (input.year % 400 == 0)) dpm=29;
62     printf("Day:");
63     scanf("%d",&input.day);
64
65     /*
66     Iteratively forcing the user to give a valid day.
67     Keynotes: the variable dpm now stores the days in the given month,days will never be less than 1
68     */
69     while(input.day>dpm || input.day<1)
70     {
71         printf("Please Enter a valid Day!!\nDay:");
72         scanf("%d",&input.day);
73     }
74     printf("\nEntered date is: ");
75     display(input);
76
77     //incrementing day by one and storing on to result variable
78     result.day = ++input.day;
79     result.month = input.month;
80     result.year = input.year;
81     /*
82     checking is result.day is greater than day limit.
83     for example if the user enter the date as 31/1/2021,
84     when we increment day by one, the result will be 32/1/2021
85     which is wrong. So we are now checking is the result.day is greater than the dpm

```

```

86     if yes we have to reset the day to 1 and increment month by 1 : 1/2/2021
87     */
88     if(result.day>dpm)
89     {
90         result.day=1;
91         /*
92         after incrementing by one month, again we are checking is the month is greater
93         than 12 if yes we have to increment year by one,
94         for example if the user enters :31/12/2021
95         incrementing day will result in :32/12/2021
96         then we reset day and increment month :01/13/2021
97         which is also wrong so we are resetting month and incrementing year
98         :01/01/2022
99         */
100        if(++result.month>12)
101        {
102            result.month =1;
103            result.year++;
104        }
105    }
106    printf("After incrementing by one day:");
107    display(result);
108
109    return 0;

```