

Using Structures to store Student Data and output filtered data using Functions

Rajashekhar Channur — IIT2021048
BTech. Information Technology
Department
IIIT Allahabad

Atharva Gadekar — IIT2021049
BTech. Information Technology
Department
IIIT Allahabad

Amisha Singh — IIT2021050
BTech. Information Technology
Department
IIIT Allahabad

Agreema Dhiman — IIT2021051
BTech. Information Technology
Department
IIIT Allahabad

Abstract—Create a structure to specify data on students as given below: Roll number, Name, Department, Course, and Year of joining. Assume that there are not more than 450 students in college. (a) Write a function to print the names of all students who joined in the last 3 years. (b) Write a function to print the data of a student whose roll numbers are divisible by 4.

I. INTRODUCTION

In C, we can divide a large program into the basic building blocks known as function. The function contains the set of programming statements enclosed by curly braces. A function can be called multiple times to provide reusability and modularity to the C program. In other words, we can say that the collection of functions creates a program. We have used functions in this program to input and display data in accordance with the given question. We are also asked to use Structures in this program. A structure is a user-defined data type in C/C++. A structure creates a data type that can be used to group items of possibly different types into a single type. We have defined a structure “students” and then defined 3 functions: one to take data from the user, second to find the names of all students who joined in the last 3 years from the data and third to print the data of a student whose roll numbers are divisible by 4. Finally, these functions are used to give this information as output.

II. ALGORITHM DESIGN

- Firstly we use the preprocessor directive `#include <stdio.h>` to use functions like **printf**, **gets** etc.
- We declare the data type *structstudents*, containing 5 structure members-
 - **rno** (which is an integer data type)
 - **name**(which is a string containing 30 characters)
 - **dept** (which is a string containing 30 characters)
 - **course** (which is a string containing 30 characters)
 - **year** (which is an integer data type)
- We declare an array of structure variables **data**, in which each element of the array will represent a structure variables. The array of structure variables **data** has a

size of 450. 450 here is the maximum number of student entries that can be stored in the program

- We declare the integer variable **n** which is given the initial value of 0.
- We declare and define the function **input** using *voidinput()*. Its declaration indicates that after completing its execution, it does not return a value. This ‘does not return any value’ is indicated using the keyword **void**.
- In the definition of the function **input**, we have commanded to print the statement “Enter the total number of students whose data you wish to input”. The **scanf** statement will assign integer variable **n** the value provided by the user.
- Then, a for loop is used where the loop counter **i** is declared as an integer and is given the initial value of 1. The condition for the loop to be executed is that the value of **i** should be less than or equal to **n**. Each time the body of the loop is executed, the value of **i** is increased by 1. When $i = 1$, the **printf**(“\n Enter the Roll Number of person No. %d: ”, **i**) displays the statement “Enter the roll number of the person 1” and similarly for all the **printf** statements in the body of the loop. We use **scanf** to assign the integer values provided by the user to the integer variables and gets for a string variable. To access any member of a structure, we use the member access operator (**.**). The member access operator is coded as a period between the structure variable name and the structure member that we wish to access. Thus, for the roll number of person 1 i.e when $i = 1$, **rno** for the structure variable *data[0]* is assigned a value equal to the value provided by the user. Similarly the input provided by the user for various members of the structure variable *data[0]* are stored in them. (We have used *getchar()* before gets statement for the name to prevent *gets()* from getting skipped. Using *getchar()* would lead to itself getting skipped instead of **gets**)
This completes the definition of the function to receive input from the user.
- Now we declare a function named *last3years* to print the

data of students who have joined in the last 3 years. We declare an integer **k** and initialize its value to 0. Now we have initialized a for loop which will run **n** times where **n** is the number of students whose data we have stored. The for loop contains an **'if'** statement where we check if the year of joining is greater than or equal to 2019. If the if statement is true, **k** is given the value 1 and the data of the students are printed. If the statement is not true, **k** is given the value 0 and the message "*No Entries Found!*" is printed on the screen.

- Now we have declared another function named *rnodiv*, where we will write a function to print the data of the students whose roll numbers are divisible by 4. We will initiate a for loop which will run **n** times. Inside the for loop if statements are written. The condition for the if statement is given by using the modulus (%) operator. We will check if the remainder is 0 when the roll number is divided by 4. If it is found to be true, the data of that particular student is printed on the screen. If the condition is not true, a message will be displayed on the screen - "*No Entries found!*".
- After defining all functions we call these functions under the *intmain()* function.

III. CONCLUSION

So, we have seen that the given question can be solved by defining functions for each part of the question. We have also used structures in the program to store data of students under one name in a block of memory. This helps in understanding how functions and structures work in C. In the end, we called all the functions which ended our program.

IV. REFERENCES

- "Let Us C" by Yashvant Kanetkar
- Tutorial Gateway - Functions in C - [Click to Visit](#)
- GeekForGeeks - Structures in C - [Click to Visit](#)

V. ACKNOWLEDGEMENTS

We would like to express our special thanks to our respected professor Dr. Mohammed Javed and our Teaching Assistant for giving us this assignment which helped us to get more familiar with the concept of Structures and Functions.

VI. APPENDIX (OUTPUT)

```
Enter the Total Number of students whose data you wish to input : 4
Enter the Roll Number of Person No. 1 : 48
Enter the Name of Person No. 1 : Rajashekhar Channur
Enter the Department of Person No. 1 : IT
Enter the Course of Person No. 1 : Btech
Enter the Year of Admission of Person No. 1 : 2020
Enter the Roll Number of Person No. 2 : 49
Enter the Name of Person No. 2 : Atharva Gadekar
Enter the Department of Person No. 2 : IT
Enter the Course of Person No. 2 : Mtech
Enter the Year of Admission of Person No. 2 : 2018
Enter the Roll Number of Person No. 3 : 50
Enter the Name of Person No. 3 : Amisha Singh
Enter the Department of Person No. 3 : ECE
Enter the Course of Person No. 3 : Btech
Enter the Year of Admission of Person No. 3 : 2021
Enter the Roll Number of Person No. 4 : 51
Enter the Name of Person No. 4 : Agreeema Dhiman
Enter the Department of Person No. 4 : ECE
Enter the Course of Person No. 4 : Mtech_

Enter the Roll Number of Person No. 4 : 51
Enter the Name of Person No. 4 : Agreeema Dhiman
Enter the Department of Person No. 4 : ECE
Enter the Course of Person No. 4 : Mtech
Enter the Year of Admission of Person No. 4 : 2019

List of Students who have been admitted in the last 3 years
Name           : Rajashekhar Channur
Roll No.       : 48
Department     : IT
Course         : Btech
Year of Admission : 2020
Name           : Amisha Singh
Roll No.       : 50
Department     : ECE
Course         : Btech
Year of Admission : 2021
Name           : Agreeema Dhiman
Roll No.       : 51
Department     : ECE
Course         : Mtech
Year of Admission : 2019

List of Students whose Roll No. is divisible by 4
Name           : Rajashekhar Channur
Roll No.       : 48
Department     : IT
Course         : Btech
Year of Admission : 2020
-----
Process exited after 191.2 seconds with return value 0
Press any key to continue . . .
```

VII. CODE

```
1 #include <stdio.h>
2
3 struct students
4 {
5     int rno;
6     char name[30];
7     char dept[30];
8     char course[30];
9     int year;
10 };
11
12 struct students data[450];
13 int n = 0;
14
15 void input ()
16 {
17     printf ("\nEnter the Total Number of students whose data you wish to input : ");
18     scanf ("%d", &n);
19
20     for (int i = 1; i <= n; i++)
21     {
22         printf ("\nEnter the Roll Number of Person No. %d : ", i);
23         scanf ("%d", &data[i - 1].rno);
24         printf ("\nEnter the Name of Person No. %d : ", i);
25         getchar ();
26         gets (data[i - 1].name);
27         printf ("\nEnter the Department of Person No. %d : ", i);
28         gets (data[i - 1].dept);
29         printf ("\nEnter the Course of Person No. %d : ", i);
30         gets (data[i - 1].course);
31         printf ("\nEnter the Year of Admission of Person No. %d : ", i);
32         scanf ("%d", &data[i - 1].year);
33     }
34 };
35
36 void last3years ()
37 {
38
39     int k = 0;
40
41     printf ("\n");
42     printf ("\nList of Students who have been admitted in the last 3 years");
43     printf ("\n");
44     for (int i = 0; i < n; i++)
45     {
46         if (data[i].year >= 2019)
47         {
48             k=1;
49             printf ("\nName           : %s", data[i].name);
50             printf ("\nRoll No.       : %d", data[i].rno);
51             printf ("\nDepartment      : %s", data[i].dept);
52             printf ("\nCourse         : %s", data[i].course);
53             printf ("\nYear of Admission : %d", data[i].year);
54             printf ("\n");
55         }
56     }
57
58     if(k==0)
59     {
60         printf ("\nNo Entries found!");
61     }
62 };
63
64 void rnodiv ()
65 {
66
67     int k = 0;
68
69     printf ("\n");
```

```

70 printf ("\nList of Students whose Roll No. is divisible by 4");
71 printf ("\n");
72
73 for (int i = 0; i < n; i++)
74 {
75     if (data[i].rno % 4 == 0)
76     {
77         k=1;
78         printf ("\nName           : %s", data[i].name);
79         printf ("\nRoll No.       : %d", data[i].rno);
80         printf ("\nDepartment    : %s", data[i].dept);
81         printf ("\nCourse       : %s", data[i].course);
82         printf ("\nYear of Admission : %d", data[i].year);
83         printf ("\n");
84     }
85 }
86
87 if(k==0)
88 {
89     printf ("\nNo Entries found!");
90 }
91 };
92
93
94 int main ()
95 {
96     input ();
97     last3years ();
98     rnodiv ();
99     return 0;
100 };

```