**Lab 2b (** [**https://github.com/Its-Masoom/su21-lab-starter**](https://github.com/Its-Masoom/su21-lab-starter) **)**

**Task1:** Write a program to store and print the name, ID, and age of a student using structures.

I have written the following program

#include <stdio.h>

struct student {

char name[50];

int id;

int age;

};

int main() {

struct student s1;

printf("Enter student name: ");

scanf("%s", s1.name);

printf("Enter student id: ");

scanf("%d", &s1.id);

printf("Enter student age: ");

scanf("%d", &s1.age);

printf("\nStudent name is %s\n", s1.name);

printf("Student id is %d\n", s1.id);

printf("Student age is %d\n", s1.age);

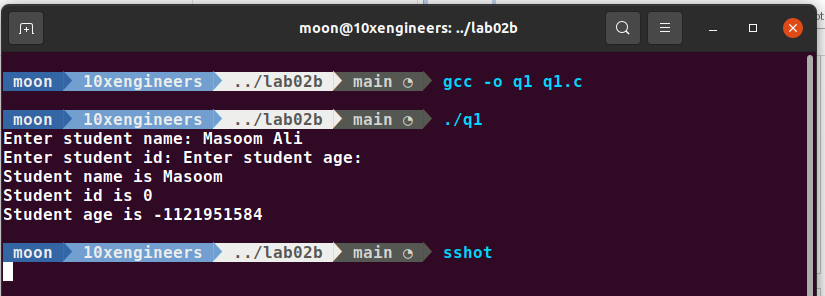
return 0;

}

**Output:**

****

But the **issue** with this program is that when there is a space in the student’s name, the program does not work as expected as shown below.



This is because scanf() stops reading as soon as it encounter a space. To resolve this issue we can use fgets() statement instead of scanf(). The revised code is:

#include <stdio.h>

#include <string.h>

struct student {

char name[50];

int id;

int age;

};

int main() {

struct student s1;

printf("Enter student name: ");

// fgets is used so that space in the name can also be read

fgets(s1.name, sizeof(s1.name), stdin);

// To remove extra line that prints after student name because fgets also takes input of a new line

s1.name[strcspn(s1.name, "\n")] = '\0';

printf("Enter student id: ");

scanf("%d", &s1.id);

printf("Enter student age: ");

scanf("%d", &s1.age);

printf("\nStudent name is %s\n", s1.name);

printf("Student id is %d\n", s1.id);

printf("Student age is %d\n", s1.age);

return 0;

}

And the output when there is a space in the name is:



**Task2:**

Here is my program:

// Program that returns an integer value by using a static inline function

#include <stdio.h>

static inline int multiply (int a, int b){

int c = a\*b;

return c;

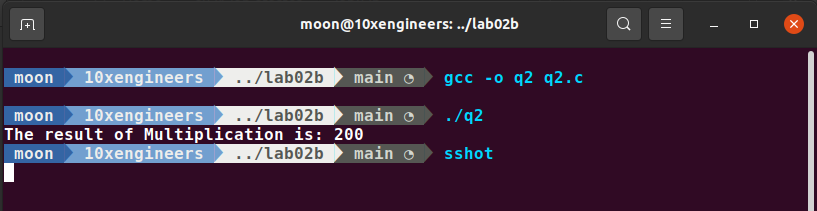
}

int main(){

printf("The result of Multiplication is: %d", multiply(10,20));

}

**Output:**



**Note:**

* Try to use inline when function size is small
* It is upto the compiler to use it inline or not even after we had specified it.
* As we have make our function static so this multiply function can only be used in the same file q2.c and it cannot be used in any other file.