

DAILY ASSESSMENT REPORT

Date:	19 June 2020	Name:	Gagan M K
Course:	C Programming	USN:	4AL17EC032
Topic:	<ul style="list-style-type: none">Structures & UnionsMemory Management	Semester & Section:	6 th sem & 'A' sec
GitHub Repository:	Alvas-education-foundation/Gagan-Git		

FORENOON SESSION DETAILS

Image of session

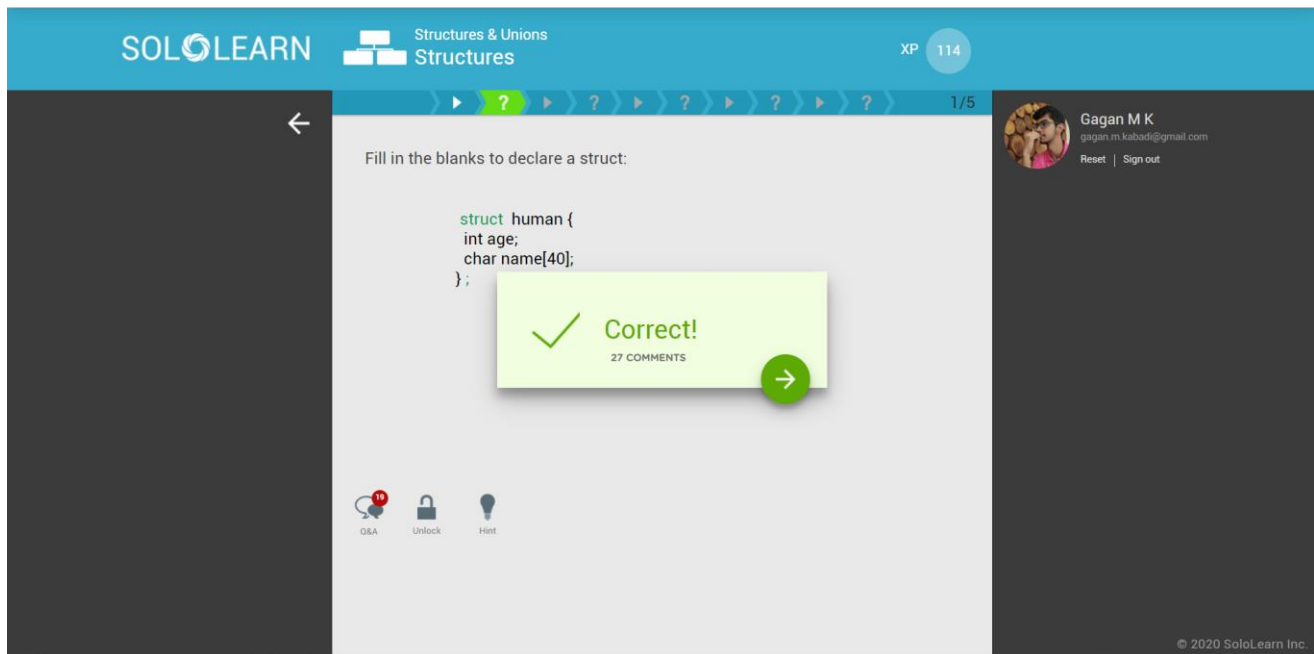
The screenshot shows a SoloLearn lesson interface. At the top, the SoloLearn logo is on the left, and the current topic 'Structures & Unions' and 'Structures' are in the center. A progress bar shows 1/5 lessons completed, with a green arrow indicating the current lesson. On the right, the user's profile 'Gagan M K' is visible with a 'Reset' and 'Sign out' option. The main content area is titled 'Structures' and contains the following text: 'A **structure** is a **user-defined data type** that groups related variables of different data types. A **structure declaration** includes the keyword **struct**, a **structure tag** for referencing the **structure**, and curly braces {} with a list of variable declarations called **members**. For example:

```
struct course {  
    int id;  
    char title[40];  
    float hours;  
};
```

 Below the code, it states: 'This struct statement defines a new data type named **course** that has three members. Structure members can be of any data type, including basic types, strings, arrays, pointers, and even other structures, as you will learn in a later lesson.' A yellow callout box says: 'Do not forget to put a semicolon after **structure** declaration. A **structure** is also called a **composite** or **aggregate** data type. Some languages refer to structures as **records**.' At the bottom right of the content area, it says '59 COMMENTS'. A green arrow button is at the bottom right of the page. The footer shows '© 2020 SoloLearn Inc.'

Report – Report can be typed or hand written for up to two pages.

C Programming:



- A structure is a user-defined data type available in C that allows to combining data items of different kinds. Structures are used to represent a record.
- Defining a structure: To define a structure, you must use the struct statement. The struct statement defines a new data type, with more than or equal to one member. The format of the struct statement is as follows:

```
struct [structure name]  
{  
    member definition;  
    member definition;  
    ...  
    member definition;  
};
```

Union:

- A union is a special data type available in C that allows storing different data types in the same memory location. You can define a union with many members, but only one member can contain a value at any given time. Unions provide an efficient way of using the same memory location for multiple purposes.
- Defining a Union: To define a union, you must use the union statement in the same way as you did while defining a structure. The union statement defines a new data type with more than one member for your program. The format of the union statement is as follows:

```
union [union name]
{
    member definition;
    member definition;
    ...
    member definition;
};
```

Similarities between Structure and Union:

- Both are user-defined data types used to store data of different types as a single unit.
- Their members can be objects of any type, including other structures and unions or arrays. A member can also consist of a bit field.
- Both structures and unions support only assignment = and sizeof operators. The two structures or unions in the assignment must have the same members and member types.
- A structure or a union can be passed by value to functions and returned by value by functions. The argument must have the same type as the function parameter. A structure or union is passed by value just like a scalar variable as a corresponding parameter.
- '.' operator is used for accessing members.

Error Handling :

- Error handling features are not supported by C programming, which is known as exception handling in C++ or in other OOP (Object Oriented Programming) languages.
- However, there are few methods and variables available in C's header file `error.h` that is used to locate errors using return values of the function call. In C, the function return `NULL` or `-1` value in case of any error, and there is a global variable `errno` which sets the error code/number.
- Hence, the return value can be used to check error while programming.

/* Divided By zero Error i.e. Exception*/

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


```
void main() {
    int ddend = 60;
    int dsor = 0;
    int q;

    if( dsor == 0){
        fprintf(stderr, "Division by zero! Exiting...\n");
        getch();
        exit(-1);
    }
    q = ddend / dsor;
    fprintf(stderr, "Value of quotient : %d\n", q);
    getch();
    exit(0);
}
```


File Handling in C:


- So far the operations using C program are done on a prompt / terminal which is not stored anywhere. But in the software industry, most of the programs are written to store the information fetched from the program.
- One such way is to store the fetched information in a file. Different operations that can be performed on a file are:
 1. Creation of a new file (fopen with attributes as "a" or "a+" or "w" or "w++")
 2. Opening an existing file (fopen)
 3. Reading from file (fscanf or fgets)
 4. Writing to a file (fprintf or fputs)
 5. Moving to a specific location in a file (fseek, rewind)
 6. Closing a file (fclose)

“Attended Webinar on “**Skills for Work - Negotiation skills**” conducted by Cambridge Assessment”

**Cambridge Assessment**
English



RECONNECT


**Cambridge**
English Qualifications

**BEC**
Business

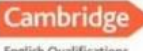
BEC from Cambridge- exam orientation app


<https://www.cambridgeenglish.org/in/>



**Cambridge Assessment**
English


RECONNECT

**Cambridge**
English Qualifications

**BEC**
Business

Saying yes

1. YES: when you are in complete agreement
2. YES: when you agree but have doubts
3. YES: when you reluctantly agree
4. YES: in appreciation of a suggestion or solution
5. YES: when you are speculating and thinking about what it might lead to



Picture source: body language and facial expressions www.bodylanguageuniversity.com

