

# SE1012 — Programming Methodology

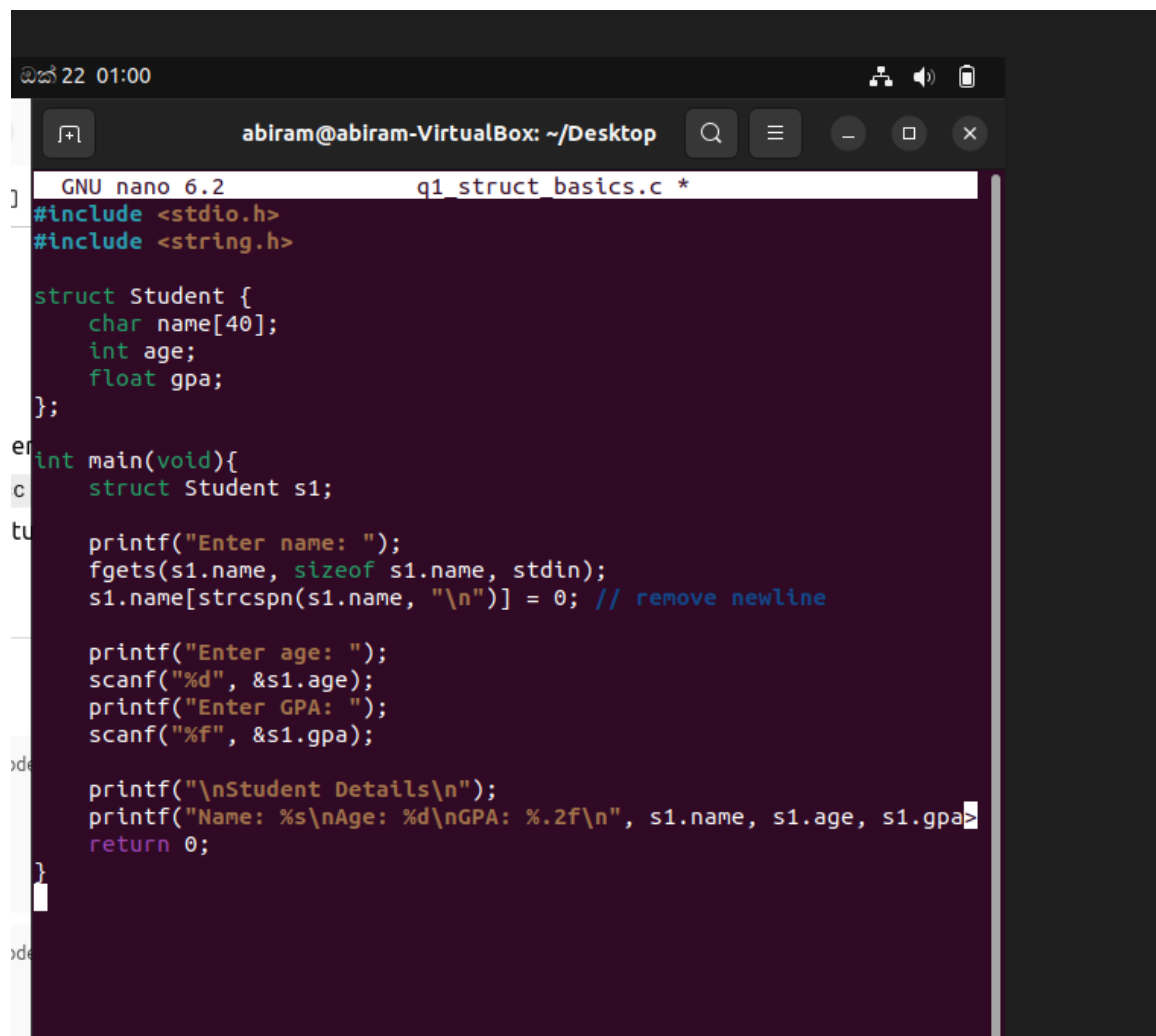
## Lab 09: String

Student Name: S. Abiram

IT Number: IT23669826

Q1)

Code



The screenshot shows a terminal window with a dark background. At the top, the title bar reads 'abiram@abiram-VirtualBox: ~/Desktop'. Below the title bar, the terminal shows the GNU nano 6.2 editor editing a file named 'q1\_struct basics.c'. The code is as follows:

```
GNU nano 6.2          q1_struct basics.c *
#include <stdio.h>
#include <string.h>

struct Student {
    char name[40];
    int age;
    float gpa;
};

int main(void){
    struct Student s1;

    printf("Enter name: ");
    fgets(s1.name, sizeof s1.name, stdin);
    s1.name[strcspn(s1.name, "\n")] = 0; // remove newline

    printf("Enter age: ");
    scanf("%d", &s1.age);
    printf("Enter GPA: ");
    scanf("%f", &s1.gpa);

    printf("\nStudent Details\n");
    printf("Name: %s\nAge: %d\nGPA: %.2f\n", s1.name, s1.age, s1.gpa);
    return 0;
}
```

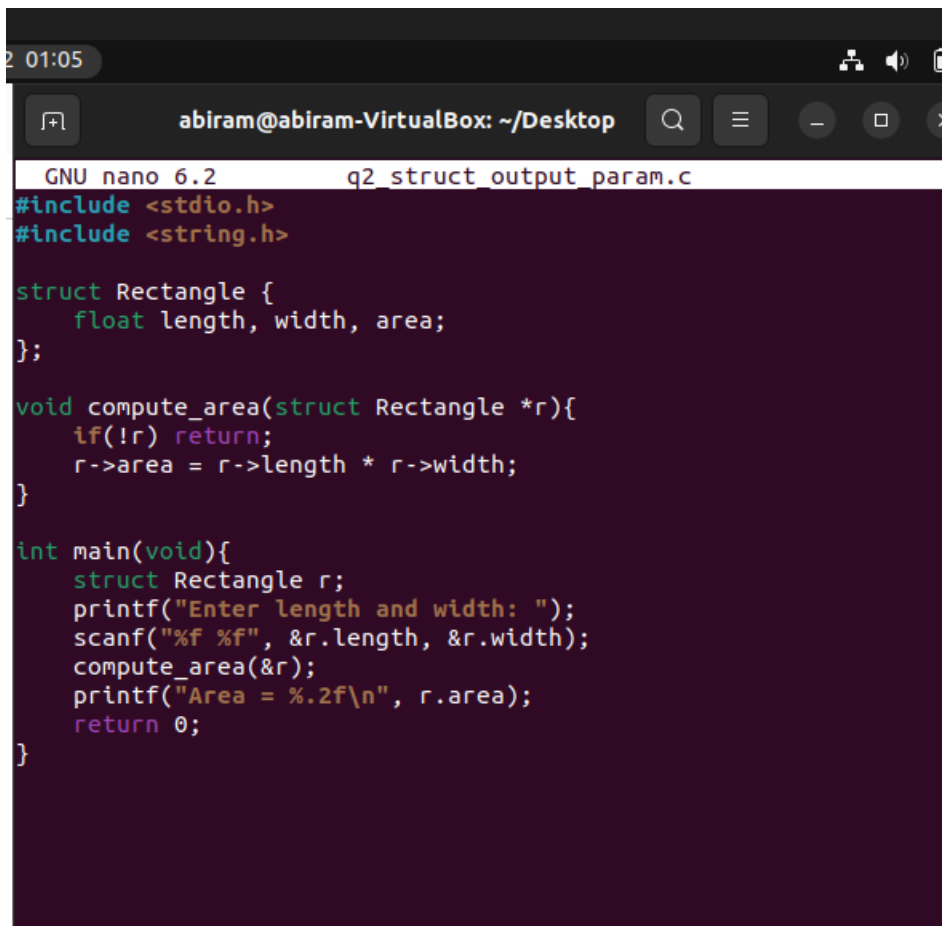
Output

Q2)

Code

```
2 01:05
abiram@abiram-VirtualBox: ~/Desktop
abiram@abiram-VirtualBox:~/Desktop$ nano q2_struct_output_param.c
abiram@abiram-VirtualBox:~/Desktop$ gcc q2_struct_output_param.c
-o q2 && ./q2
Enter length and width: 20
3
Area = 60.00
abiram@abiram-VirtualBox:~/Desktop$
```

Output



The image shows a terminal window with a dark background. At the top, a status bar displays '2 01:05' on the left and system icons on the right. Below this, a terminal title bar shows 'abiram@abiram-VirtualBox: ~/Desktop' with search, menu, and window control buttons. The main area of the terminal is titled 'GNU nano 6.2' and 'q2\_struct\_output\_param.c'. It contains C code for a program that calculates the area of a rectangle. The code includes headers for `<stdio.h>` and `<string.h>`, defines a `Rectangle` struct with `length`, `width`, and `area` as float members, and implements a `compute_area` function and a `main` function. The `main` function prompts the user for length and width, calls `compute_area`, and prints the resulting area with two decimal places.

```
GNU nano 6.2          q2_struct_output_param.c
#include <stdio.h>
#include <string.h>

struct Rectangle {
    float length, width, area;
};

void compute_area(struct Rectangle *r){
    if(!r) return;
    r->area = r->length * r->width;
}

int main(void){
    struct Rectangle r;
    printf("Enter length and width: ");
    scanf("%f %f", &r.length, &r.width);
    compute_area(&r);
    printf("Area = %.2f\n", r.area);
    return 0;
}
```

Q3)

Code

```
2 01:06
abiram@abiram-VirtualBox: ~/Desktop
GNU nano 6.2 q3_struct_return.c *
#include <stdio.h>
#include <string.h>

struct Employee {
    int id;
    char name[50];
    double salary;
};

struct Employee read_employee(void){
    struct Employee e;
    printf("Enter ID: ");
    scanf("%d", &e.id);
    getchar(); // clear buffer
    printf("Enter Name: ");
    fgets(e.name, sizeof e.name, stdin);
    e.name[strcspn(e.name, "\n")] = 0;
    printf("Enter Salary: ");
    scanf("%lf", &e.salary);
    return e;
}

int main(void){
    struct Employee e1 = read_employee();
    printf("\nEmployee Info\n");
    printf("ID: %d\nName: %s\nSalary: %.2f\n", e1.id, e1.name, e1.salary);
    return 0;
}
```

Output

```
2 01:07
abiram@abiram-VirtualBox: ~/Desktop
abiram@abiram-VirtualBox:~/Desktop$ nano q3_struct_return.c
abiram@abiram-VirtualBox:~/Desktop$ gcc q3_struct_return.c -o q3
&& ./q3
Enter ID: 324
Enter Name: qwwe
Enter Salary: 23454

Employee Info
ID: 324
Name: qwwe
Salary: 23454.00
abiram@abiram-VirtualBox:~/Desktop$
```

Q4)

Code

```
2 01:08
abiram@abiram-VirtualBox: ~/Desktop
GNU nano 6.2      q4 array of structs.c *
#include <stdio.h>
#include <string.h>

#define N 3
struct Book {
    char title[60];
    char author[40];
    float price;
};

int main(void){
    struct Book books[N];
    for(int i=0; i<N; ++i){
        printf("\nBook %d title : ", i+1);
        fgets(books[i].title, sizeof books[i].title, stdin);
        books[i].title[strcspn(books[i].title, "\n")] = 0;

        printf("Author : ");
        fgets(books[i].author, sizeof books[i].author, stdin);
        books[i].author[strcspn(books[i].author, "\n")] = 0;

        printf("Price (LKR): ");
        scanf("%f", &books[i].price);
        getchar(); // clear newline
    }

    printf("\nBook List:\n");
    for(int i=0; i<N; ++i)
        printf("%d %-20s by %-15s Rs. %.2f\n", i+1,
            books[i].title, books[i].author, books[i].price);
    return 0;
}

^G Help      ^O Write Out ^W Where Is ^K Cut      ^T Execute
```

Output

```
2 01:09
abiram@abiram-VirtualBox: ~/Desktop
abiram@abiram-VirtualBox:~/Desktop$ nano q3_struct_return.c
abiram@abiram-VirtualBox:~/Desktop$ gcc q3_struct_return.c -o q3
&& ./q3
Enter ID: 324
Enter Name: qwwe
Enter Salary: 23454

Employee Info
ID: 324
Name: qwwe
Salary: 23454.00
abiram@abiram-VirtualBox:~/Desktop$ nano q4_array_of_structs.c
abiram@abiram-VirtualBox:~/Desktop$ gcc q4_array_of_structs.c -o
q4 && ./q4

Book 1 title : er3
Author : ewrsdsf
Price (LKR): 2345

Book 2 title : sgweg
Author : swe
Price (LKR): 3421

Book 3 title : dvsdvs
Author : sdvsdvs
Price (LKR): 321

Book List:
1) er3                by ewrsdsf                Rs. 2345.00
2) sgweg              by swe                  Rs. 3421.00
3) dvsdvs             by sdvsdvs             Rs. 321.00
abiram@abiram-VirtualBox:~/Desktop$
```

Q5)

Code

2 01:10

```
abiram@abiram-VirtualBox: ~/Desktop
GNU nano 6.2 q5_parallel_vs_struct.c *
#include <stdio.h>
#include <string.h>

#define N 3

// Using parallel arrays
void parallel_arrays(void){
    char names[N][40];
    int marks[N];
    printf("\n[Parallel Arrays]\n");
    for(int i=0;i<N;++i){
        printf("Name: ");
        scanf(" %39[^\n]", names[i]);
        printf("Marks: ");
        scanf("%d", &marks[i]);
    }
    printf("\nResults (Parallel)\n");
    for(int i=0;i<N;++i)
        printf("%-15s %3d\n", names[i], marks[i]);
}

// Using array of structures
struct Student {
    char name[40];
    int marks;
};

void array_of_structs(void){
    struct Student s[N];
    printf("\n[Array of Structures]\n");
    for(int i=0;i<N;++i){
        printf("Name: ");
        scanf(" %39[^\n]", s[i].name);
        printf("Marks: ");
        scanf("%d", &s[i].marks);
    }
}

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
```