



**Developed By**

## **PSP[20ES104] COURSE PROJECT REPORT ON**

**“ Restaurant Billing Using C ”**

**Developed By :**

**Hall Ticket Number :**

**Student Name :**

2203A51741,

G.Aashish Reddy

2203A52209,

B.Rahul thrinethra

2203A51740

T. Koushik

Under the Guidance of:

Mr. P. PRAMOD KUMAR, M.Tech.(Ph.D)

Senior Assistant Professor

Submitted to Ms.Triveni Sadala

Department Computer Science and Artificial Intelligence

***SR UNIVERSITY***

Ananthasagar(V), Hasanparthy(M), Hanamkonda(Dist.) – 506371

[www.sru.edu.in](http://www.sru.edu.in)

June 2023

Department of Computer Science and Artificial Intelligence

## CERTIFICATE

This is to certify that the PSP course project report entitled “ Title of the project” is a record of bonafide work carried out by the student(s) B.Rahul Thrinethra ,G.Aashish Reddy & T.Koushik&bearing roll number(s) 2203A52209 ,2203A1741 & 2203A51740 of Computer Science and Artificial Intelligence department during the academic year 2022-23.

Supervisor

( Ms.Triveni Sadala)

## INDEX

Sl. No	Title	PageNo.
1.	Problem statment	1
2.	Concepts(String & Structure)	2 to 4
3.	Final Code	5 to 8

### **Problem Statement:**

The problem statement is to write a C program that prints a restaurant bill using strings and structures. The program should take input from the user for the items ordered, their quantities, and their prices. It should then calculate the total cost and print a bill displaying the details of the order along with the total amount to be paid.

The program should have the following features:

1. Prompt the user to enter the number of items to be ordered.
2. Prompt the user to enter the number of items to be ordered.
3. Use a structure to store the item details (name, quantity, and price).
4. Calculate the total cost by multiplying the quantity and price for each item and summing up the costs of all items.
5. Display the itemized bill, including the item name, quantity, price per item, and total cost for each item.
6. Finally, display the total amount to be paid by the customer.

## Concepts :

we have used C program in that we used only 2 concepts they are;

1. String

2. Structure

- these are the concepts we used to print the booking medical appointment

What is C?

C programming is a computer programming language that was originally developed in the early 1970s by Dennis Ritchie at Bell Labs. It is a general-purpose, procedural programming language that is widely used for developing a wide range of applications, from small utility programs to large-scale operating systems.

C programming provides a structured approach to writing software by using a set of well-defined syntax rules and standard libraries. It allows programmers to manipulate data, perform calculations, and control the flow of execution using variables, functions, and control structures like loops and conditional statements.

C is known for its efficiency and low-level control, making it suitable for systems programming and embedded systems where performance and direct hardware access are crucial. It provides features like pointers, which allow direct manipulation of memory addresses, and it supports low-level programming constructs, such as bitwise operations.

The language has influenced the development of many other programming languages and is still widely used today. It serves as the foundation for many operating systems, including Unix, Linux, and Windows. C has a relatively simple syntax, which makes it easier to learn compared to some other languages, and it remains an essential language for programmers, particularly those involved in system-level development and performance-critical application

## String :

A string in C is an array of characters terminated by a null character '\0'. It is used to represent a sequence of characters. Strings can be declared as character arrays, and various string manipulation functions are available in the standard C library to work with strings. Some commonly used string functions include strcpy(), strcat(), strlen(), strcmp(), and scanf().

Here's an example of declaring and using a string in C:

```
#include <stdio.h>

int main() {
    char greeting[10] = "Hello";
    printf("%s\n", greeting);

    char name[20];
    printf("Enter your name: ");
    scanf("%s", name);
    printf("Hello, %s!\n", name);

    return 0;
}
```

## Structure:

A structure in C is a user-defined data type that allows you to combine different types of variables under a single name. It is used to create complex data structures that can hold related information. A structure declaration defines the layout of the data, and variables of that structure type can then be created to hold the data.

Here's an example of declaring and using a structure in C:

```
#include <stdio.h>

struct student {
    char name[20];
    int age;

    float gpa;
};

int main() {
    struct student s1;

    printf("Enter student name: ");
    scanf("%s", s1.name);

    printf("Enter student age: ");
    scanf("%d", &s1.age);

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

    printf("Student Information:\n");
    printf("Name: %s\n", s1.name);
    printf("Age: %d\n", s1.age);
    printf("GPA: %.2f\n", s1.gpa);
    Return0; }
```

Finial code :

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

```
#define MAX_MENU_ITEMS 10
#define MAX_ITEM_NAME_LENGTH 50
```

```
typedef struct {
    char name[MAX_ITEM_NAME_LENGTH];
    float price;
} MenuItem;
```

```
typedef struct {
    char customerName[MAX_ITEM_NAME_LENGTH];
    int itemCount;
    MenuItem items[MAX_MENU_ITEMS];
    float total;
} Bill;
```

```
void initializeBill(Bill *bill) {
    printf("Enter customer name: ");
    fgets(bill->customerName, sizeof(bill->customerName), stdin);
    bill->customerName[strcspn(bill->customerName, "\n")] = '\0'; // Remove
trailing newline

    bill->itemCount = 0;
    bill->total = 0.0;
```



```
}
```

```
void addItem(Bill *bill, const char *name, float price) {  
    if (bill->itemCount >= MAX_MENU_ITEMS) {  
        printf("Maximum item count reached.\n");  
        return;  
    }  
}
```

```
MenuItem item;  
strncpy(item.name, name, sizeof(item.name) - 1);  
item.name[sizeof(item.name) - 1] = '\0'; // Ensure null-terminated string  
item.price = price;
```

```
bill->items[bill->itemCount++] = item;  
bill->total += price;  
}
```

```
void printBill(const Bill *bill) {  
    int i;  
    printf("\nCustomer: %s\n", bill->customerName);  
    printf("-----\n");  
  
    for ( i = 0; i < bill->itemCount; i++) {  
        printf("%-30s $%.2f\n", bill->items[i].name, bill->items[i].price);  
    }  
  
    printf("-----\n");  
    printf("Total:                $%.2f\n", bill->total);  
}
```

```
}
```

```
int main() {
```

```
    Bill bill;
```

```
    initializeBill(&bill);
```

```
    printf("\nMENU:\n");
```

```
    printf("1. Burger      $4.99\n");
```

```
    printf("2. Pizza      $8.99\n");
```

```
    printf("3. Salad      $6.49\n");
```

```
    printf("4. Pasta      $9.99\n");
```

```
    printf("5. Ice Cream   $3.49\n");
```

```
    int choice;
```

```
    do {
```

```
        printf("\nEnter item number (0 to exit): ");
```

```
        scanf("%d", &choice);
```

```
        getchar(); // Consume the newline character
```

```
        switch (choice) {
```

```
            case 1:
```

```
                addItem(&bill, "Burger", 4.99);
```

```
                break;
```

```
            case 2:
```

```
                addItem(&bill, "Pizza", 8.99);
```

```
                break;
```

```
            case 3:
```

```
                addItem(&bill, "Salad", 6.49);
```

```
                break;
```

```
            case 4:
```

```
        addItem(&bill, "Pasta", 9.99);
        break;
    case 5:
        addItem(&bill, "Ice Cream", 3.49);
        break;
    case 0:
        break;
    default:
        printf("Invalid choice.\n");
    }
} while (choice != 0);

printBill(&bill);

return 0;
}
```