VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi-590 018, Karnataka

Project Report
on

"HOSPITAL MANAGEMENT SYSTEM"

Submitted in partial fulfillment of the requirements for the award of the degree of Bachelor of Engineering
in
Computer Science & Engineering

Submitted by

USN	Name
1BI20CS003	ABHINAV KUMAR
1BI20CS086	PRADDYUMNA KANDALKAR
1BI20CS057	DHEERAJ KATOCH
1BI20CS050	DARSHAN PASARGI
1BI20CS189	VIVEK GUPTA

For Academic year 2021-22

Under the Guidance of
Prof. ANJINI NAGESH
Assistant Professor
Department of CS&E, BIT
Bengaluru-560004

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
BANGALORE INSTITUTE OF TECHNOLOGY
K.R. Road, V.V. Pura, Bengaluru-560 004

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi-590 018, Karnataka

BANGALORE INSTITUTE OF TECHNOLOGY

Bengaluru-560 004

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Certificate

This is to certify that the implementation of **DATA STRUCTURES AND APPLICATION MINI PROJECT (18CS32)** entitled "**HOSPITAL MANAGEMENT SYSTEM**" has been successfully completed by

Name	USN
ABHINAV KUMAR	**1BI20CS003**
PRADDYUMNA KANDALKAR	**1BI20CS086**
DHEERAJ KATOCH	**1BI20CS057**
DARSHAN PASARGI	**1BI20CS050**
VIVEK GUPTA	**1BI20CS189**

of III semester B.E. for the partial fulfillment of the requirements for the Bachelor's degree in Computer Science & Engineering of the Visvesvaraya Technological University during the academic year 2021-2022.

In charge :

Prof. ANJINI NAGESH

Assistant Professor

Department of CS&E,

Bangalore Institute of Technology, Bangalore

Dr. J. Girija.

Professor and Head of Department

CS&E, BIT

Abstract

- The HOSPITAL MANAGEMENT is a simple project which is designed in **C language**.
- The coding of this project is done in such a way that the user feels enthralling and blessed to have come to our hospital with such good Faculties and Doctors with so user friendly Paperwork.
- This project is coded in such a way that it's very **user-friendly**. This project is developed in the IDE called as **Visual Studio Code** with the help of **GCC compiler**.
- When you run the application will be asked to enter certain details of patient.

Acknowledgement

We take this opportunity to acknowledge all the people who have helped us wholeheartedly in every stage of this project. We would like to express our sincere gratitude to Mrs. ANJINI NAGESH and our honorable principal Dr Aswath M.U. of Bangalore Institute of Technology for their valuable guidance and support in completing our project in Data Structure and Application on **HOSPITAL MANGEMENT SYSTEM.**

Your valuable guidance and suggestions helped us in various phases of the completion of this project.

Finally, as a team, we would also like to appreciate each one of us for their support and coordination in the completion of this project. We hope we will achieve more in our future endeavors.

Index

Chapter 1: Introduction

Chapter 2: Problem Statement and Objective

Chapter 3: System Requirements

Chapter 4: Flowchart with explanation

Chapter 5: Design

Chapter 6: Implementation

Chapter 7: Testing and Results

Chapter 8: Applications

Chapter 9: Conclusion and Future work

Chapter 10: References

CHAPTER-1

INTRODUCTION

1. 1 About Hospital Management System:

Hospital management system is a computer system that helps manage the information related to health care and aids in the job completion of health care providers effectively. They manage the data related to all departments of healthcare such as,

- Financial
- Inpatient
- Outpatient
- Materials
- Nursing
- Pharmaceutical
- Neurology
- Cardiology etc.

HMS came into the picture of hospital management as early as 1960 and have ever since been evolving and synchronizing with the technologies while modernizing healthcare facilities. In today's world, the management of healthcare starts from the hands of the patients through their mobile phones and facilitates the needs of the patient.

CHAPTER-2

PROBLEM STATEMENT AND OBJECTIVE

2.1 Problem Statement:

This project is aimed to automate the hospital management system. The purpose of the project entitled as HOSPITAL MANAGEMENT SYSTEM is to computerize the Front Office Management of Hospital to develop software which is user friendly, simple, fast, and cost – effective.

It deals with the collection of patients information, diagnosis details, etc. Traditionally, it was done manually.

The main function of the system is to register and store patient details and doctor details and retrieve these details as and when required, and also to manipulate these details meaningfully.

This function of Hospital Management Information System deals with registering the new Patient by giving unique Identification Number to the Patient. This number is unique throughout the System for identifying the patient. Simultaneous update and changes are made to the databases. Identification number is also provided to doctor to retrieve and to change doctor details.

The software is used by administrator or receptionist in the hospital. The software is secured by username and password, accessed by administrator or receptionist of the hospital.

2.2 Objectives:

Design a system for better patient care.

Reduce hospital operating costs.

Provide MIS (Management Information System) report on demand to management for better decision making.

Better co-ordination among the different departments.

Provide top management a single point of control.

CHAPTER-3

SYSTEM REQUIREMENTS

3.1 System requirements:

3.1.1 The minimum requirements are: -

3.1.1.1 Intel Core i5 or i7 processor

3.1.1.2 Full HD resolution, ideally 1920×1080

3.1.1.3 8GB of RAM

3.1.2 IDE used: -

VS CODE

Visual Studio Code is a code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

Also, we used Fedora IDE.

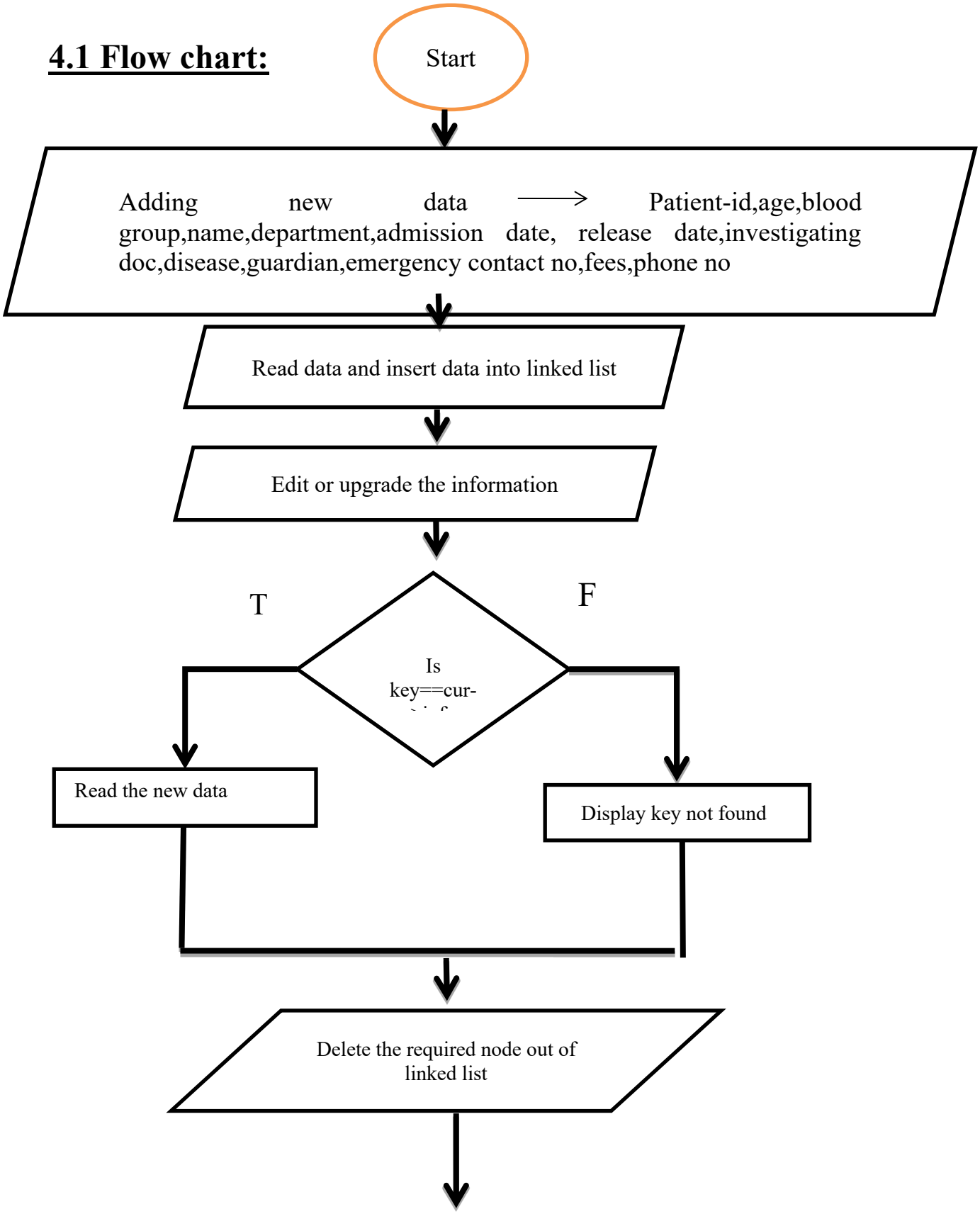
3.1.3 Compiler used: -

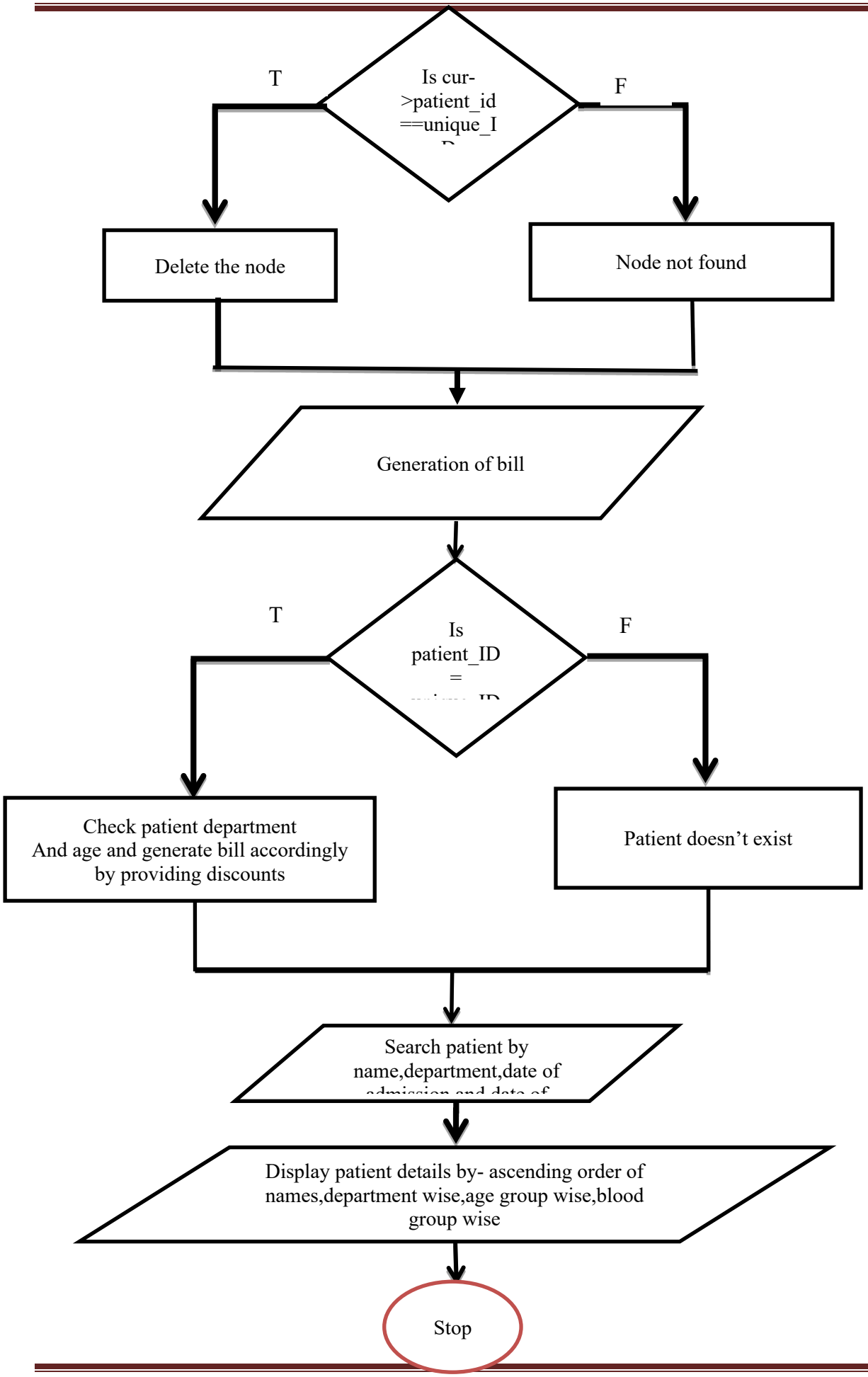
MinGW is a compiler system based on the GNU GCC and Binutils projects that compiles and links code to be run on Win32 (Windows) systems. It provides C, C++ and Fortran compilers plus other related tools. 'MinGW' refers to the "Minimalist GNU for Windows" project.

CHAPTER-4

FLOWCHART WITH EXPLANATION

4.1 Flow chart:



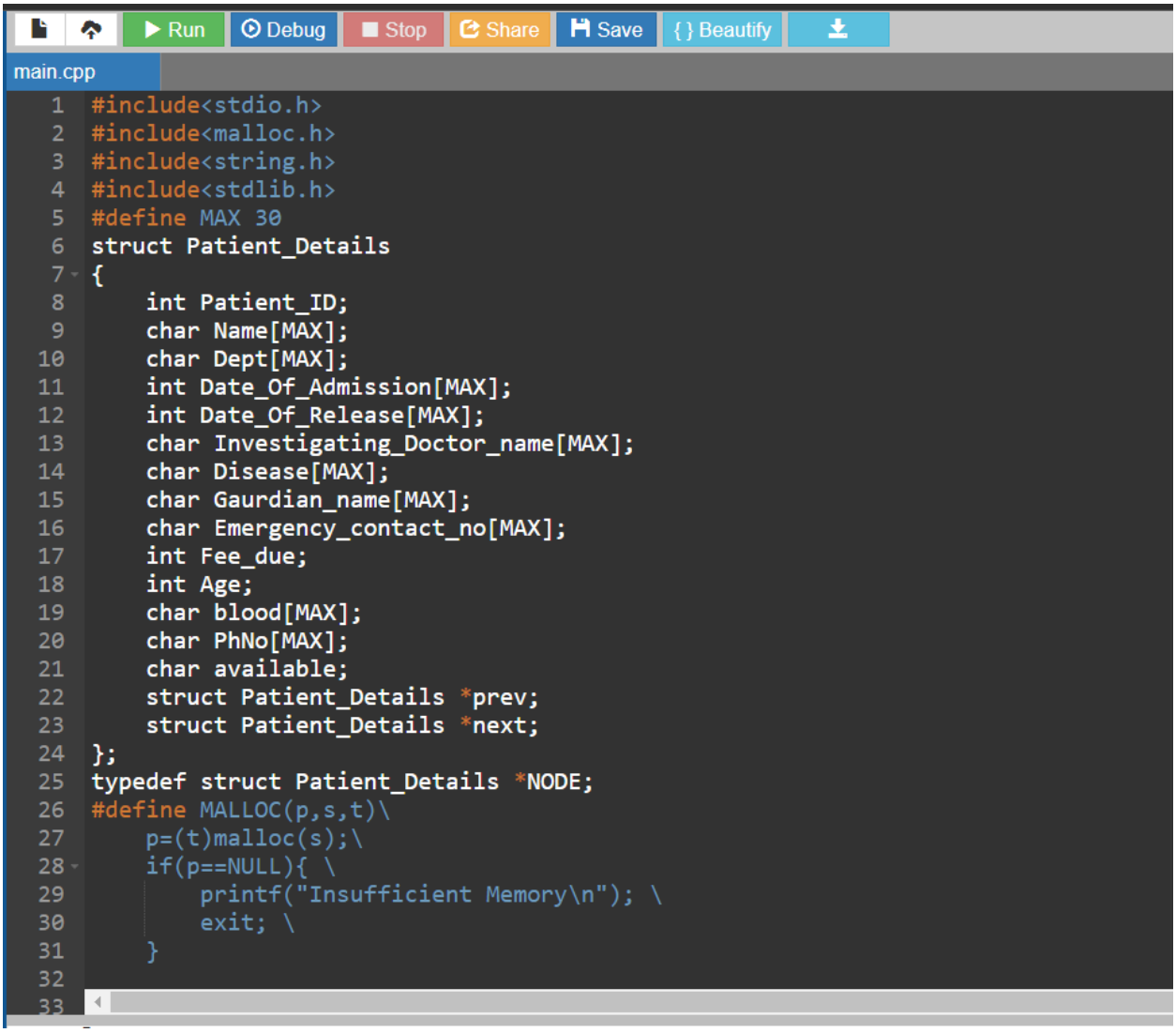

Is cur->patient_id ==unique_I
T
F
Delete the node
Node not found
Generation of bill
Is patient_ID =
T
F
Check patient department And age and generate bill accordingly by providing discounts
Patient doesn't exist
Search patient by name,department,date of
Display patient details by- ascending order of names,department wise,age group wise,blood group wise
Stop

CHAPTER-6

IMPLEMENTATION

6.1 Defining a Linked List

We define a Linked List by mentioning the parameters it's going to hold, i.e., all different fields of information.




```c
#include<stdio.h>
#include<malloc.h>
#include<string.h>
#include<stdlib.h>
#define MAX 30
struct Patient_Details
{
    int Patient_ID;
    char Name[MAX];
    char Dept[MAX];
    int Date_Of_Admission[MAX];
    int Date_Of_Release[MAX];
    char Investigating_Doctor_name[MAX];
    char Disease[MAX];
    char Gaurdian_name[MAX];
    char Emergency_contact_no[MAX];
    int Fee_due;
    int Age;
    char blood[MAX];
    char PhNo[MAX];
    char available;
    struct Patient_Details *prev;
    struct Patient_Details *next;
};
typedef struct Patient_Details *NODE;
#define MALLOC(p,s,t)\
    p=(t)malloc(s);\
    if(p==NULL){ \
        printf("Insufficient Memory\n"); \
        exit; \
    }
```

6.2 Searching of Patients Information

We have provided search option by various parameters to search making it convenient for both the medical staff and an individual user.

```
void search(NODE first)
{
    int choice;
    NODE temp;
    int c =0;
    printf("Enter your preference to Search a  Patient in the Hospital\n");
    printf("1.Name\t 2.Department \t 3.Date of Admission \t 4.Date of Release \n");
    printf("Enter your choice \n");
    scanf("%d",&choice);
    switch(choice)
    {
    case 1:
        if(first == NULL)
            printf("No patients admitted!\n");
        else{
            temp = first->next;
            printf("Enter Name \n");
            scanf("%s", name);
            while(temp!=first)
            {
                if(strcmp(name, temp->Name)==0)
                {
                    c = c+1;
                    display_individual(temp);
                }
                temp =temp->next;
            }
            if(strcmp(name, temp->Name)==0)
            {
                c = c+1;
                display_individual(temp);
```

```
        }
        if(c==0)
            printf("No patients admitted with that name! \n");
        break;
    case 2:
        if(first ==NULL)
            printf("No patients admitted!\n");
        else{
            temp = first->next;
            printf("Enter Department\n");
            scanf("%s", Dept);
            while(temp!=first)
            {
                if(strcmp(Dept, temp->Dept)==0)
                {
                    c = c+1;
                    display_individual(temp);
                }
                temp =temp->next;
            }
            if(strcmp(Dept, temp->Dept)==0)
            {
                c = c+1;
                display_individual(temp);
            }
        }
        if(c==0)
            printf("No patients admitted in that Department!\n");
        break;
    case 3:
        if(first ==NULL)
            printf("No patients admitted!\n");
```

input

main.cpp

```
    break;
case 3:
    if(first ==NULL)
        printf("No patients admitted!\n");
    else{
        temp = first->next;
        printf("Enter Patient Date of Admission\n");
        scanf("%d",&DOA[0]);
        printf("Enter Patient Month of Admission\n");
        scanf("%d",&DOA[1]);
        printf("Enter Patient Year of Admission\n");
        scanf("%d",&DOA[2]);
        while(temp!=first)
        {
            if(DOA[0] == temp->Date_Of_Admission[0] && DOA[1] == temp->Date_Of_Admission[1] && DOA[2] == temp->Date_Of_Admission[2])
            {
                c = c+1;
                display_individual(temp);
            }
            temp =temp->next;
        }
        if(DOA[0] == temp->Date_Of_Admission[0] && DOA[1] == temp->Date_Of_Admission[1] && DOA[2] == temp->Date_Of_Admission[2])
        {
            c = c+1;
            display_individual(temp);
        }
    }
    if(c==0)
        printf("No patients admitted on that date!\n");
    break;
case 4:
    if(first ==NULL)
```

main.cpp

```
case 4:
    if(first ==NULL)
        printf("No patients admitted!\n");
    else{
        temp = first->next;
        printf("Enter Patient Date of Release\n");
        scanf("%d",&DOR[0]);
        printf("Enter Patient Month of Release\n");
        scanf("%d",&DOR[1]);
        printf("Enter Patient Year of Release\n");
        scanf("%d",&DOR[2]);
        while(temp!=first)
        {
            if(DOR[0] == temp->Date_Of_Release[0] && DOR[1] == temp->Date_Of_Release[1] && DOR[2] == temp->Date_Of_Release[2])
            {
                c = c+1;
                display_individual(temp);
            }
            temp =temp->next;
        }
        if(DOR[0] == temp->Date_Of_Release[0] && DOR[1] == temp->Date_Of_Release[1] && DOR[2] == temp->Date_Of_Release[2])
        {
            c = c+1;
            display_individual(temp);
        }
    }
    if(c==0)
        printf("No patients released on that date!\n");
    break;
default : printf("Invalid Choice\n");
    break;
}
```

input

6.3 Bill Generation:

Bill Generation uses nested if -else -if Statements.

main.cpp

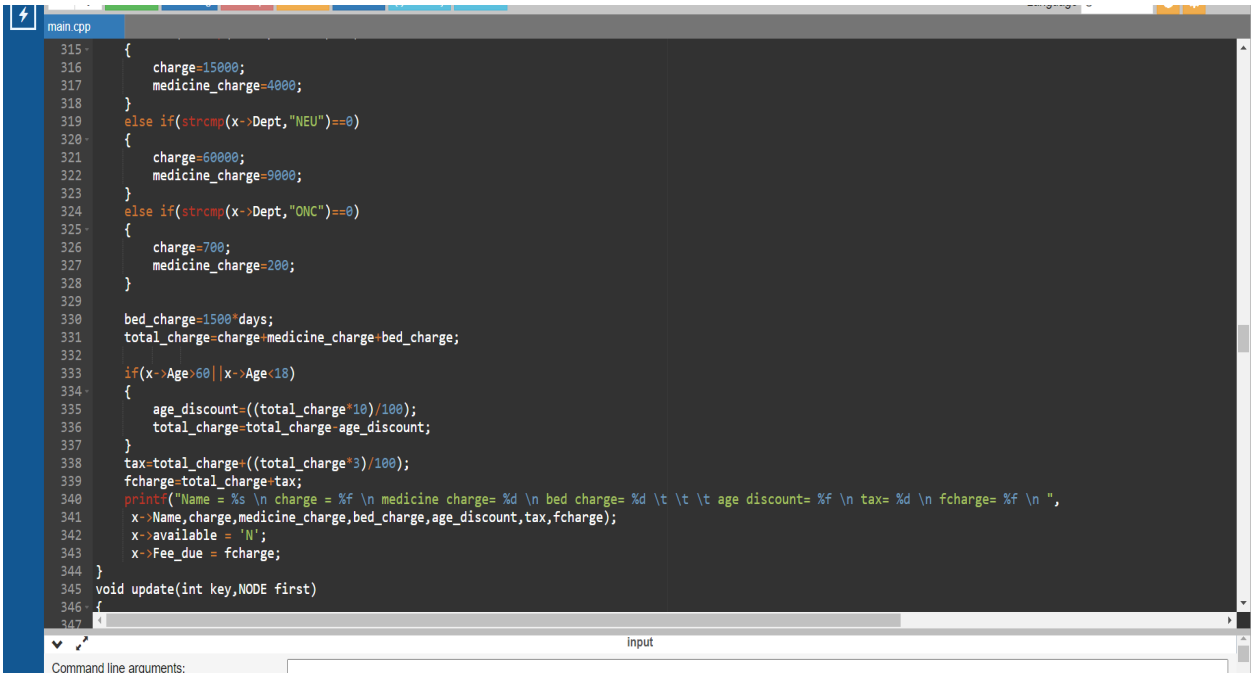
```
void bill(NODE first,int days) //days=no. of days stayed in hospital
{
    int medicine_charge,bed_charge,total_charge,tax,pid,flag;
    float age_discount;
    float charge,fcharge;
    int consul_charge=100;

    NODE x=first;
    printf("Search by Patient ID :\n");
    scanf("%d",&pid);

    if(x==NULL){
        printf("No Patient Entry\n");
    }
    else if(x->next==NULL){
        if(x->Patient_ID==pid)
            display_individual(x);
        else
            printf("Patient with given patient ID does not exist ! \n");
    }

    do{
        if(x->Patient_ID==pid)
            flag=1;
        x=x->next;
    }
    while((x!=first)&&(flag==0));

    if((flag==0)&&(x==first)){
        printf("Patient with given patient ID does not exist ! \n");
    }
    else if(strcmp(x->Dept,"AND")==0)
```

input

main.cpp

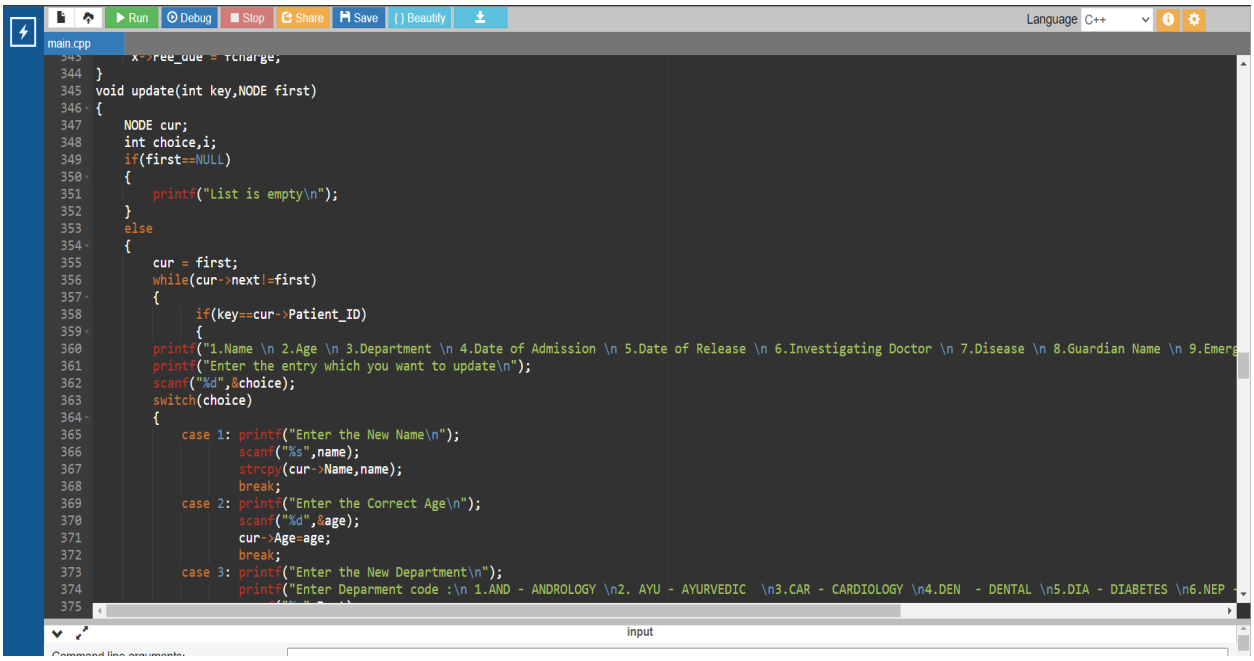
```
    if((flag==0)&&(x==first)){
        printf("Patient with given patient ID does not exist ! \n");
    }
    else if(strcmp(x->Dept,"AND")==0)
    {
        charge = 2000;
        medicine_charge=1000;
    }
    else if(strcmp(x->Dept,"AYU")==0)
    {
        charge =500;
        medicine_charge=300;
    }
    else if(strcmp(x->Dept,"CAR")==0)
    {
        charge =40000;
        medicine_charge=5000;
    }
    else if(strcmp(x->Dept,"DEN")==0)
    {
        charge =1200;
        medicine_charge=300;
    }
    else if(strcmp(x->Dept,"DIA")==0)
    {
        charge=1300;
        medicine_charge=500;
    }
    else if(strcmp(x->Dept,"NEP")==0)
    {
        charge=15000;
        medicine_charge=4000;
```

input



6.4 Editing any information:

We Have used many ways to edit any particular field of information by using Switch Cases.



main.cpp

```
                    strcpy(first->Dept,Dept);
                    break;
            case 4:
                    printf("Enter Patient Date of Admission\n");
                    scanf("%d", &DOA[0]);
                    printf("Enter Patient Month of Admission\n");
                    scanf("%d", &DOA[1]);
                    printf("Enter Patient Year of Admission\n");
                    scanf("%d", &DOA[2]);
                    for(i = 0;i<3;i++)
                    cur->Date_Of_Admission[i] = DOA[i];
                    break;

            case 5:
                    printf("Enter Patient Date of Release\n");
                    scanf("%d", &DOR[0]);
                    printf("Enter Patient Month of Release\n");
                    scanf("%d", &DOR[1]);
                    printf("Enter Patient Year of Release\n");
                    scanf("%d", &DOR[2]);
                    for(i = 0;i<3;i++)
                    cur->Date_Of_Release[i] = DOR[i];
                    break;

            case 6: printf("Enter the New Investigating Doctor\n");
                    scanf("%s", investigating_Doctor_name);
                    strcpy(cur->Investigating_Doctor_name, investigating_Doctor_name );
                    break;
            case 7: printf("Enter the New Disease Name\n");
                    scanf("%s",disease);
                    strcpy(cur->Disease,disease);
                    break;
```

input

main.cpp

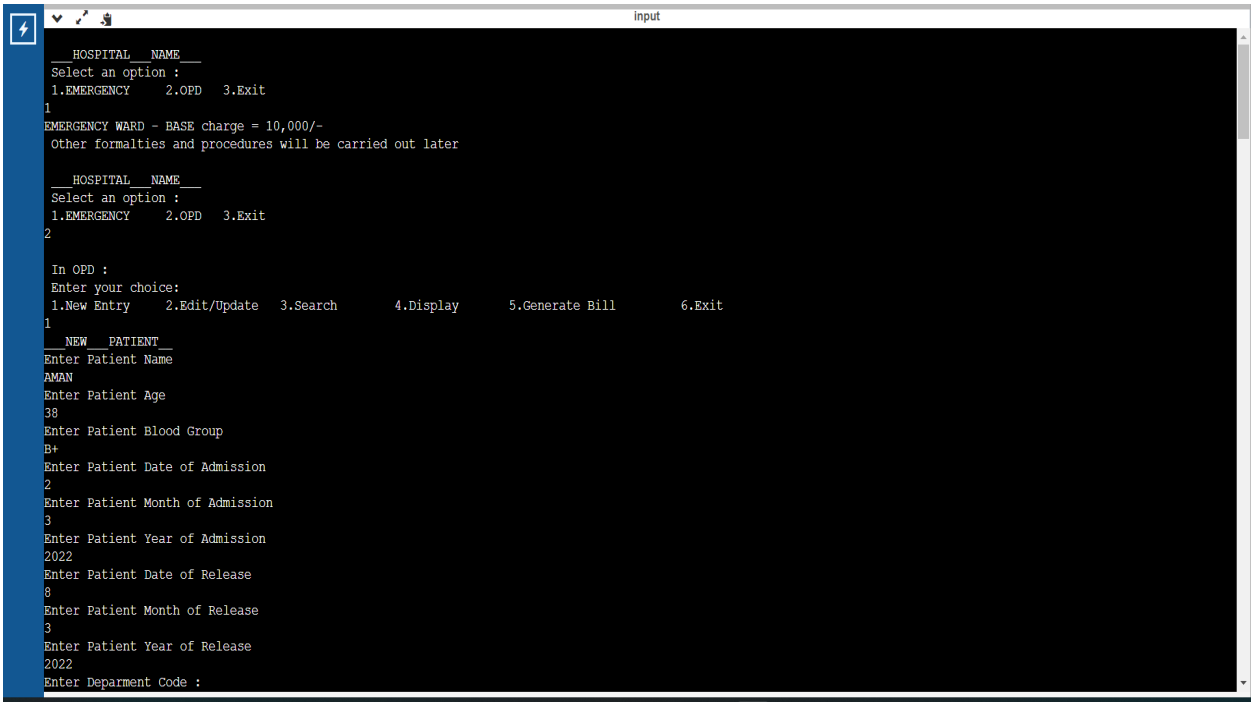
```
                    scanf("%s", investigating_Doctor_name);
                    strcpy(cur->Investigating_Doctor_name, investigating_Doctor_name );
                    break;
            case 7: printf("Enter the New Disease Name\n");
                    scanf("%s",disease);
                    strcpy(cur->Disease,disease);
                    break;
            case 8:printf("Enter the New Guardian Name\n");
                    scanf("%s", gaurdian_name);
                    strcpy(cur->Gaurdian_name, gaurdian_name);
                    break;
            case 9:printf("Enter the New Emergency Contact Number \n");
                    scanf("%s", emergency_contact_no);
                    strcpy(cur->Emergency_contact_no,emergency_contact_no);
                    break;
            case 10:printf("Enter the New Phone Number \n");
                    scanf("%s", phNo);
                    strcpy(cur->PhNo,phNo);
                    break;
            default: printf("INVALID CHOICE\n");
        }
        break;
    }
    cur = cur->next;
}
if(key==cur->Patient_ID)
            {
        printf("1.Name \n 2.Age \n 3.Department \n 4.Date of Admission \n 5.Date of Release \n 6.Investigating Doctor \n 7.Disease \n 8.Guardian Name \n 9.Emerg
        printf("Enter the Entry which you want to update\n");
        scanf("%d",&choice);
        switch(choice)
```

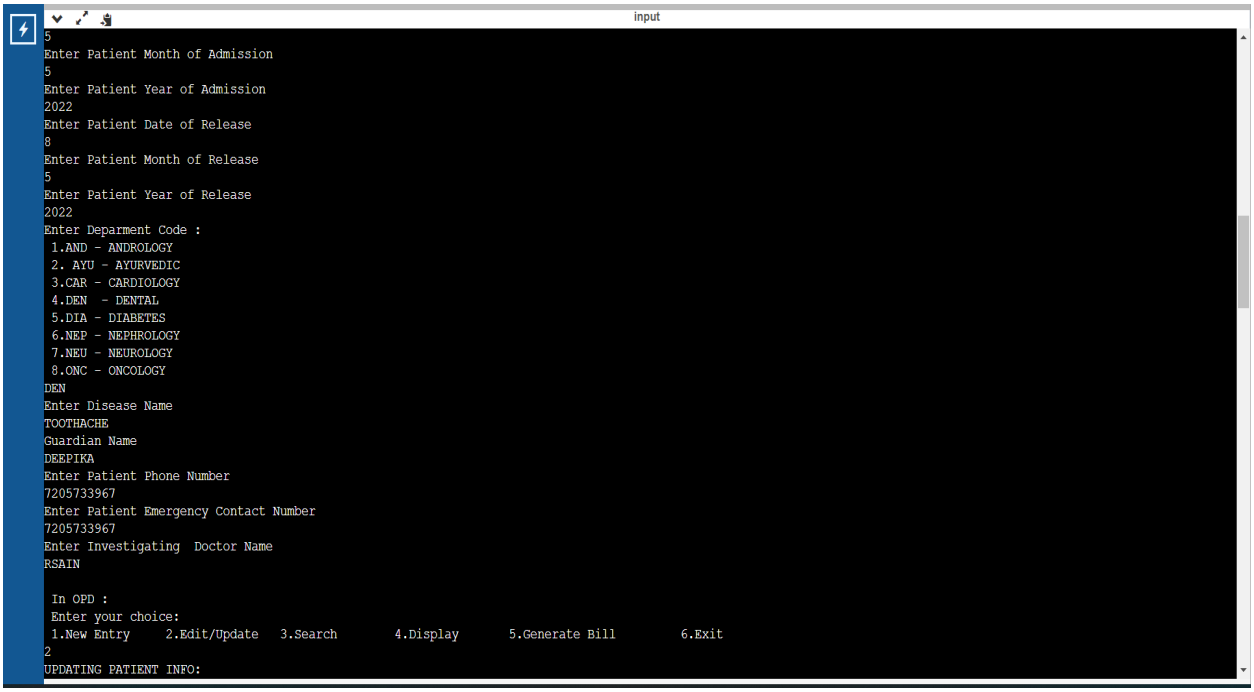
CHAPTER-7

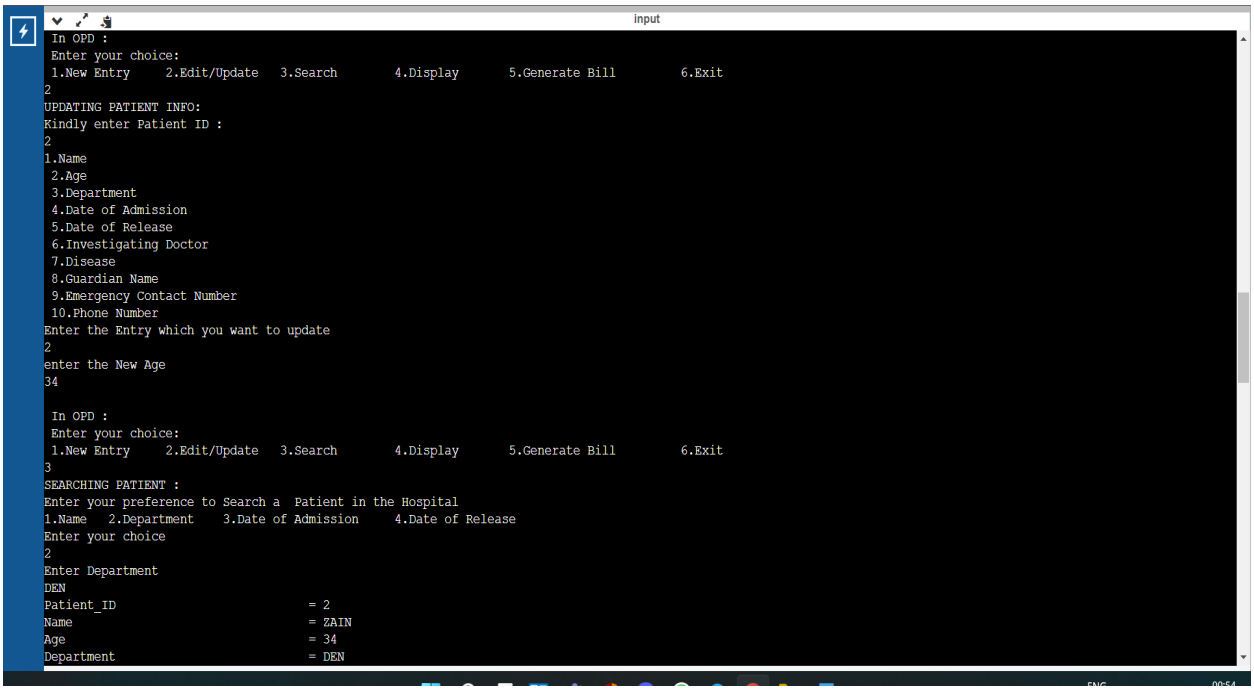
TESTING AND RESULTS

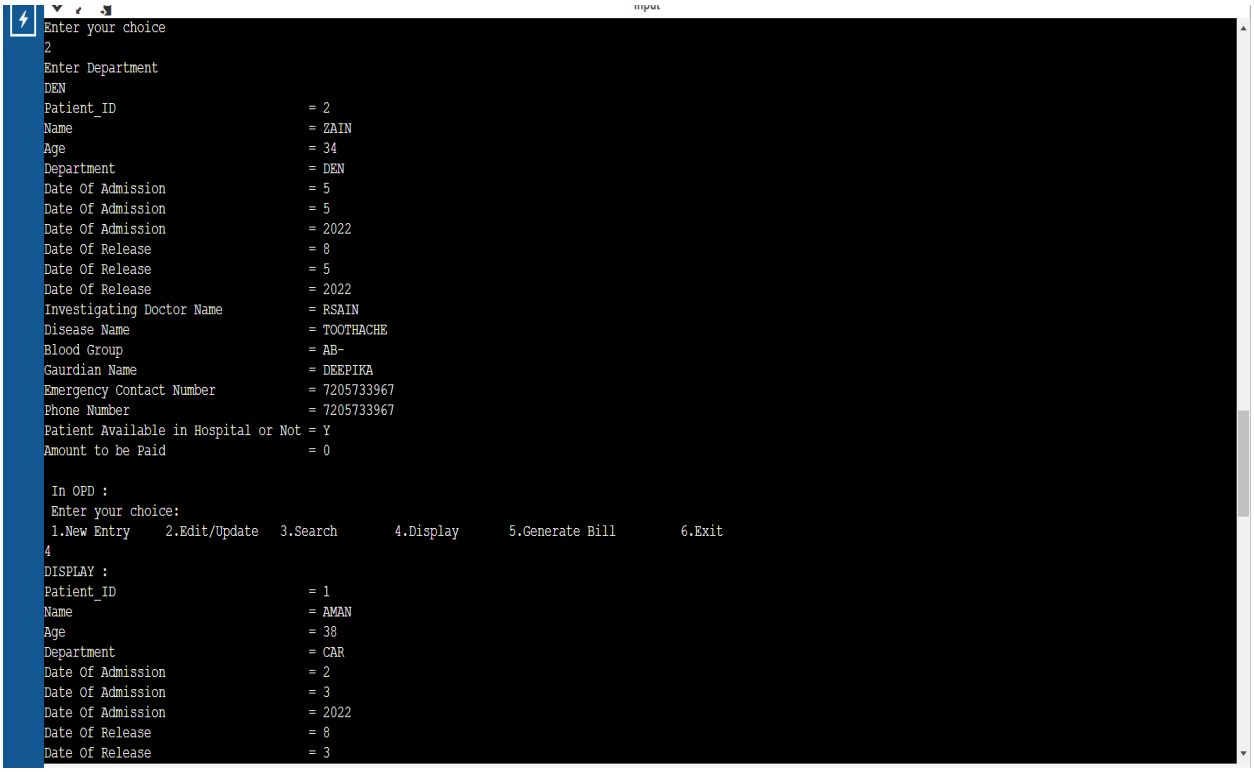
7.1 Testing:

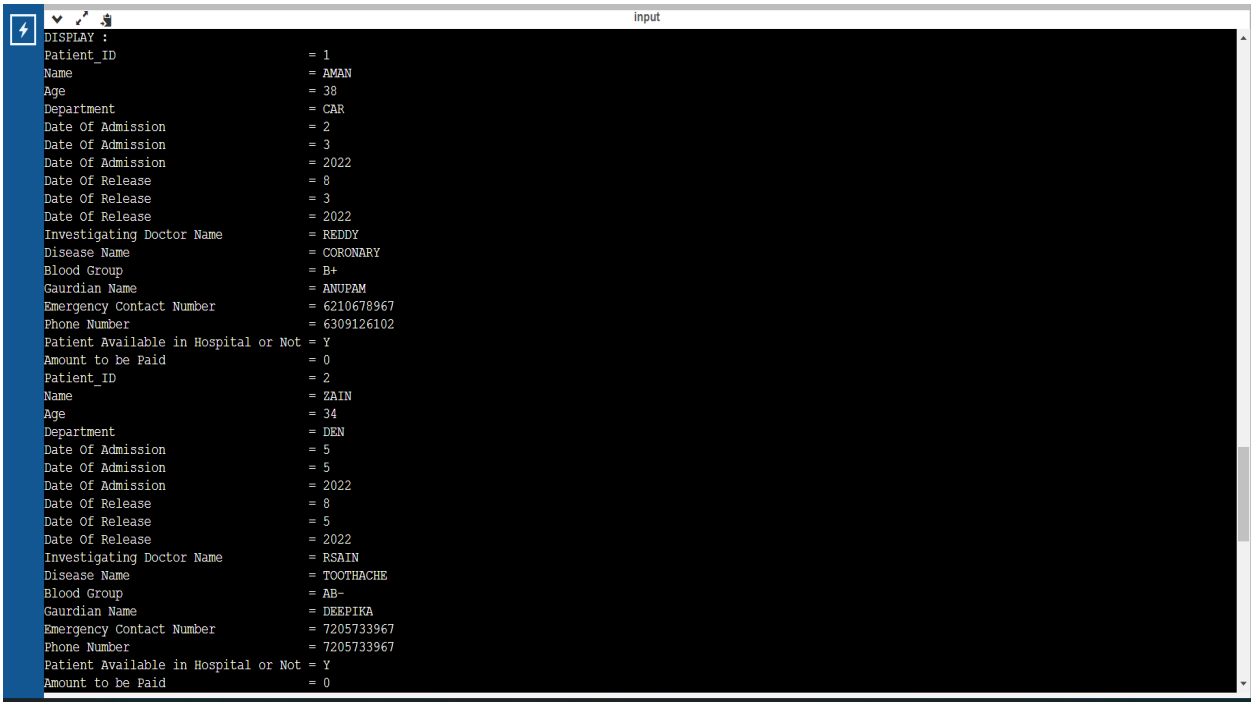
Here we have provided pictures of code running in actual time.

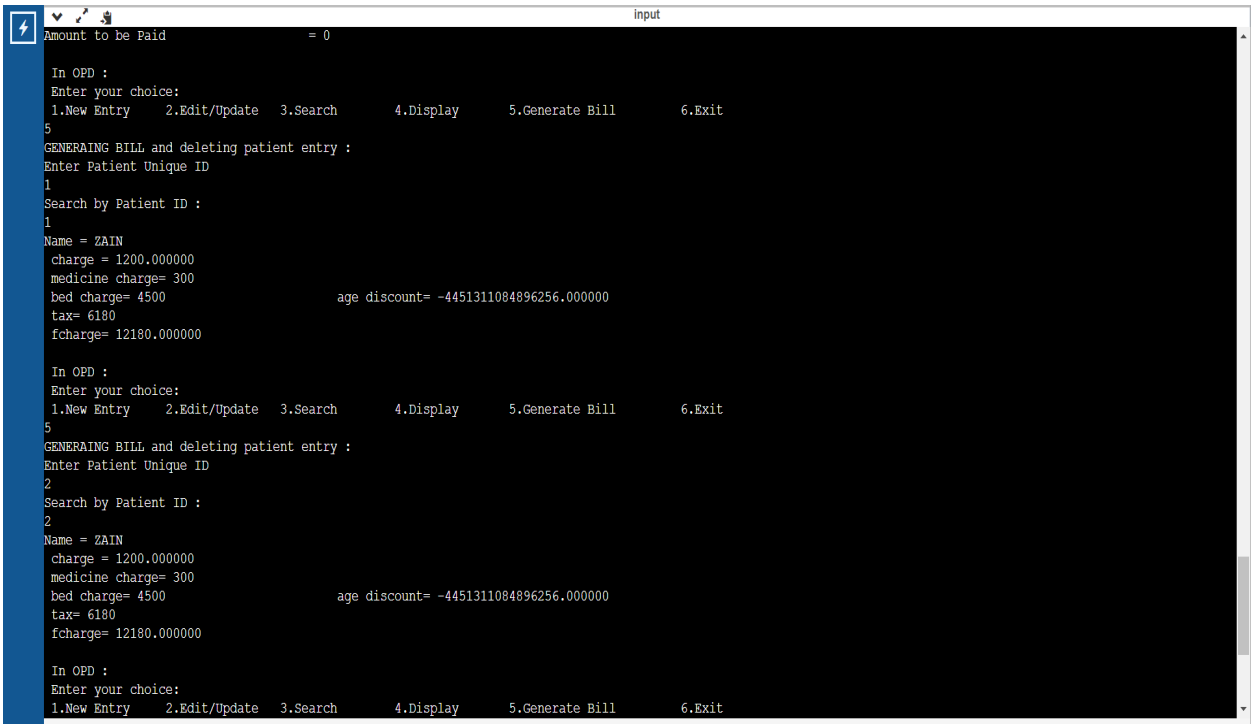


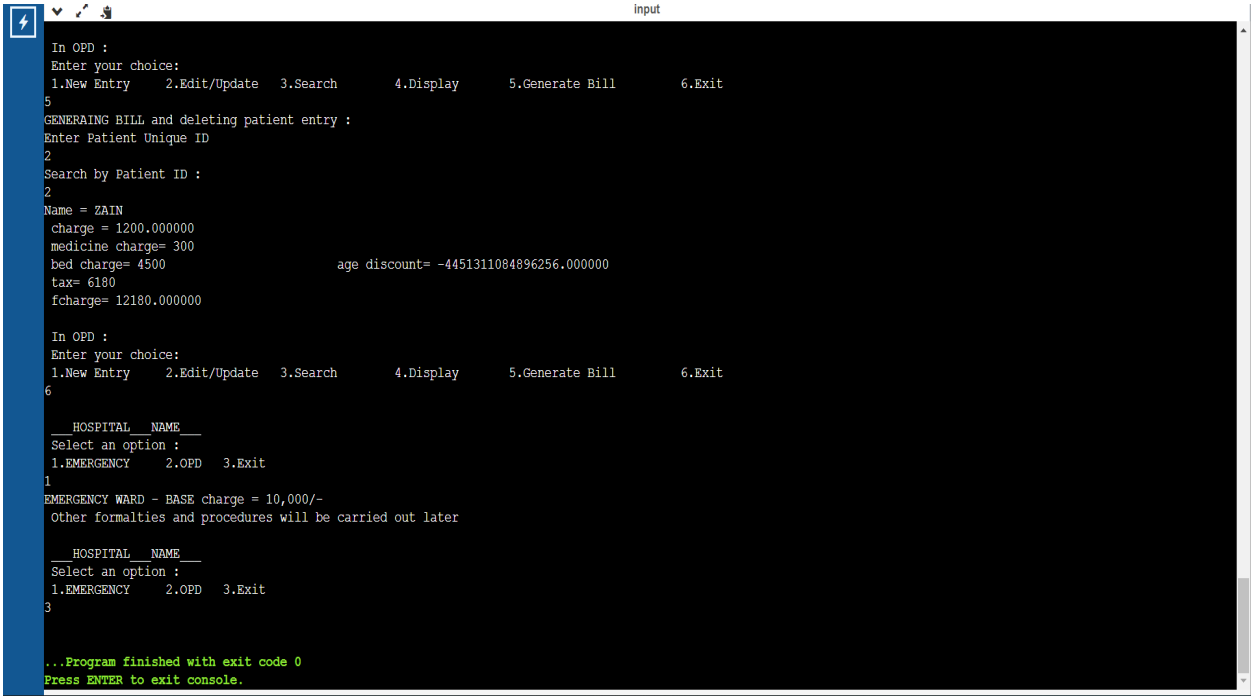



input
In OPD :
Enter your choice:
1.New Entry 2.Edit/Update 3.Search 4.Display 5.Generate Bill 6.Exit
2
UPDATING PATIENT INFO:
Kindly enter Patient ID :
2
1.Name
2.Age
3.Department
4.Date of Admission
5.Date of Release
6.Investigating Doctor
7.Disease
8.Guardian Name
9.Emergency Contact Number
10.Phone Number
Enter the Entry which you want to update
2
enter the New Age
34
In OPD :
Enter your choice:
1.New Entry 2.Edit/Update 3.Search 4.Display 5.Generate Bill 6.Exit
3
SEARCHING PATIENT :
Enter your preference to Search a Patient in the Hospital
1.Name 2.Department 3.Date of Admission 4.Date of Release
Enter your choice
2
Enter Department
DEN
Patient_ID = 2
Name = ZAIN
Age = 34
Department = DEN


Enter your choice
2
Enter Department
DEN
Patient_ID = 2
Name = ZAIN
Age = 34
Department = DEN
Date Of Admission = 5
Date Of Admission = 5
Date Of Admission = 2022
Date Of Release = 8
Date Of Release = 5
Date Of Release = 2022
Investigating Doctor Name = RSAIN
Disease Name = TOOTHACHE
Blood Group = AB-
Gaurdian Name = DEEPIKA
Emergency Contact Number = 7205733967
Phone Number = 7205733967
Patient Available in Hospital or Not = Y
Amount to be Paid = 0
In OPD :
Enter your choice:
1.New Entry 2.Edit/Update 3.Search 4.Display 5.Generate Bill 6.Exit
4
DISPLAY :
Patient_ID = 1
Name = AMAN
Age = 38
Department = CAR
Date Of Admission = 2
Date Of Admission = 3
Date Of Admission = 2022
Date Of Release = 8
Date Of Release = 3


input
DISPLAY :
Patient_ID = 1
Name = AMAN
Age = 38
Department = CAR
Date Of Admission = 2
Date Of Admission = 3
Date Of Admission = 2022
Date Of Release = 8
Date Of Release = 3
Date Of Release = 2022
Investigating Doctor Name = REDDY
Disease Name = CORONARY
Blood Group = B+
Gaurdian Name = ANUPAM
Emergency Contact Number = 6210678967
Phone Number = 6309126102
Patient Available in Hospital or Not = Y
Amount to be Paid = 0
Patient_ID = 2
Name = ZAIN
Age = 34
Department = DEN
Date Of Admission = 5
Date Of Admission = 5
Date Of Admission = 2022
Date Of Release = 8
Date Of Release = 5
Date Of Release = 2022
Investigating Doctor Name = RSAIN
Disease Name = TOOTHACHE
Blood Group = AB-
Gaurdian Name = DEEPIKA
Emergency Contact Number = 7205733967
Phone Number = 7205733967
Patient Available in Hospital or Not = Y
Amount to be Paid = 0


input
Amount to be Paid = 0
In OPD :
Enter your choice:
1.New Entry 2.Edit/Update 3.Search 4.Display 5.Generate Bill 6.Exit
5
GENERAING BILL and deleting patient entry :
Enter Patient Unique ID
1
Search by Patient ID :
1
Name = ZAIN
charge = 1200.000000
medicine charge= 300
bed charge= 4500 age discount= -4451311084896256.000000
tax= 6180
fcharge= 12180.000000
In OPD :
Enter your choice:
1.New Entry 2.Edit/Update 3.Search 4.Display 5.Generate Bill 6.Exit
5
GENERAING BILL and deleting patient entry :
Enter Patient Unique ID
2
Search by Patient ID :
2
Name = ZAIN
charge = 1200.000000
medicine charge= 300
bed charge= 4500 age discount= -4451311084896256.000000
tax= 6180
fcharge= 12180.000000
In OPD :
Enter your choice:
1.New Entry 2.Edit/Update 3.Search 4.Display 5.Generate Bill 6.Exit


input
In OPD :
Enter your choice:
1.New Entry 2.Edit/Update 3.Search 4.Display 5.Generate Bill 6.Exit
5
GENERAING BILL and deleting patient entry :
Enter Patient Unique ID
2
Search by Patient ID :
2
Name = ZAIN
charge = 1200.000000
medicine charge= 300
bed charge= 4500 age discount= -4451311084896256.000000
tax= 6180
fcharge= 12180.000000
In OPD :
Enter your choice:
1.New Entry 2.Edit/Update 3.Search 4.Display 5.Generate Bill 6.Exit
6
___HOSPITAL___NAME___
Select an option :
1.EMERGENCY 2.OPD 3.Exit
1
EMERGENCY WARD - BASE charge = 10,000/-
Other formalties and procedures will be carried out later
___HOSPITAL___NAME___
Select an option :
1.EMERGENCY 2.OPD 3.Exit
3
...Program finished with exit code 0
Press ENTER to exit console.

CHAPTER-8

APPLICATIONS

8.1 Applications:

With technology revolutionizing every industry, the health sector is not left behind. With several hospital software system developers emerging, it is imperative that any specialist get the right software.

Note that when it comes to handling patients is your health facility, certain services can be simplified with the right software.

CHAPTER-9

CONCLUSION AND FUTURE WORK

9.1 Conclusion and Future Work:

9.1.1 CONCLUSION

The code we built is a basic level code which will help the hospital management system in providing a user friendly and interactive interface for dealing with all the paper work and necessary hospital related query.

9.1.2 FUTURE WORK

Change is inevitable so are we humans, Change is Constant. We all grow, develop, and prosper So will our code and the mini project we built. In order to improve our existing project, we can further add the cash free treatment covered under several Health Insurance Policies and further add the insurance segment also in our project.

CHAPTER-10

REFERENCES

10.1 References:

10.1.1 geeksforgeeks.org

10.1.2 stackoverflow.com

10.1.3 www.tutorialspoint.com

10.1.4 https://aticleworld.com/hospital-management-system-project-in-c/