MINOR PROJECT 1

FINAL REPORT

on

HOSTEL MANAGEMENT SYSTEM

Submitted By :

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | VIBHOR JAIN | | SHASHI KUMAR | | YASH KUMAR | |  | |
|  | 500054739 | | 500053083 | | 500052876 | |  | |
|  | |  | |  | |  | |  |

***Under the guidance of***

**Ms. SHAHINA ANWARUL**

Assistant Professor

Department of Systemics,

School of Computer Science



School of Computer Science

**UNIVERSITY OF PETROLEUM AND ENERGY STUDIES Dehradun**

**2017-18**



CANDIDATES DECLARATION

We hereby certify that the project work entitled Title of Project in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science And Engineering with Specialization in open source and open standards and submitted to the Department of Informatics at School of Computer Science, University of Petroleum And Energy Studies, Dehradun, is an authentic record of our work carried out during a period from August, 2018 to December, 2018 under the supervision of Ms. SHAHINA ANWARUL

Assistant Professor,Department of Systemics, School of Computer Science

The matter presented in this project has not been submitted by me/ us for the award of any other degree of this or any other University.

VIBHOR JAIN SHASHI KUMAR YASH KUMAR

500054739 500053083 500052876

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

MS.SHAHINA ANWARUL Project Guide Assistant Professor Department of Systemics, School of Computer Science

Dr. Monit Kapoor

Head

Department of Informatics

School of Computer Science

University of Petroleum And Energy Studies

Dehradun - 248 001 (Uttarakhand )

ACKNOWLEDGEMENT

We wish to express our deep gratitude to our guide Name, for all advice, encouragement and constant support he has given us through out our project work. This work would not have been possible without his support and valuable suggestions.

We sincerely thank to our Head of the Department, Dr. Monit Kapoor, for his great support in doing our project name at SoCS.

We are also grateful to Dr. Manish Prateek Professor and Director SoCS and Dr. Kamal Bansal Dean CoES, UPES for giving us the necessary facilities to carry out our project work successfully.

We would like to thank all our friends for their help and constructive criticism during our project work. Finally we have no words to express our sincere gratitude to our parents who have shown us this world and for every support they have given us.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

**ABSTRACT**

Hostel management system is a software developed for managing activities in hostel. Even nowadays there are several hostels which operate manually and take a lot of time throughout the whole process and in Campus hostels student does not know what he is getting, so this is not a transparent system. This project deals with problem of managing a hostel and avoids problems which occur manually. All the processes in the system will be fully transparent to the user. There is increasing number of private hostels around colleges and many of them don’t use software, so this will also be useful there. So the drawbacks of the present system create the need to designing of computerized system. The program will be written entirely in C programming language and will use files to store data, instead of databases. We will provide verification using login authentication page developed in c language.We arel using bubble sort for sorting the data so that it will be in ascending order We are using binary search for searching from sorted element. It will make our code more efficient.

Keywords**:-**

Hostel management, Automation, C programming, Login authentication , sorting , searching,validation.

**(I)**

TABLE OF CONTENTS

1. Introduction 1
2. Objective 1
3. Literature Review 1-2

4 Problem statement 2

5 Flowchart 3

6 Implementation 4

6.1 Pseudocode

6.2 Output Screen 5-7

7 References 7

A APPENDIX PROJECT CODE

**LIST OF FIGURES**

1 Login authentication page

2 File menu

3 View List

4 Entry Details

1)  **INTRODUCTION**

This project is for managing the various activities in the hostel. It will manage the student information, room information, room allocation and employee details of the hostel. It is also used to generate reports of student details. It will also keep tracks of number of students in room and availability of rooms at the time of registration. The main purpose of this project is to manage the details of beds and hostel room allotment via transparent process with the choice of students. This system is designed in favor of the hostel management which helps them to save the records of the students about the rooms and other things. This system also helps students by displaying everything before the registrations. It helps them from the manual work from which it is very difficult to find the records of students and the information of about those who had left the hostels. We can improve the efficiency of the system, thus overcome the drawbacks of the existing systems. We design the system of the hostel management especially for the college hostels.

2) **Objective**

To design and develop an automated system that can make processes such as room allotment and record searching easy and error free.

Sub- Objectives:-

1. To automate each and every activity of the manual systems.
2. Maintain security of the records
3. Reduce the cost of maintenance.
4. Room provided with the choice of students.

3) **Literature Review: -**

After going through some research papers and reports on Hostel Management System, we learned that for the past few years the numbers of educational institutions are increasing rapidly. Thereby the numbers of hostels are also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who are running the hostel and software’s are not usually used in this context. This particular project deals with the problems on managing a hostel and avoids the problems which occur when carried manually [**1].**

We observed that the system has the following drawbacks:-

· More human error.

· More strength and strain of manual labor needed

· Repetition of the same procedures.

· Low security

· Data redundancy

· Difficult to handle

· Difficult to update and retrieve data

· Record keeping is difficult

· Backup data can be easily generated **[2][3].**

We found that the proposed system is very easy to operate. Speed and accuracy are the main advantages of proposed system. There is no redundancy of data. The data are stored in the computer secondary memories like hard disk, etc. it can be easily receive and used at any time. The proposed system will easily handle all the data and the work done by the existing systems. The proposed systems eliminate the drawbacks of the existing system to a great extent and it provides tight security to data **[4].**

4) **Problem Statement:-**

The hostel management today is mostly done using lots of paperwork which is a tedious task and creates room for human error.

5**) Flowchart**

Login authentication

False

True

View Menu

Options

Exit

View List

Remove

Details

Update Info

Enter Details

etai

Search Info

isValid

Enter Data

Yes

Data Entered

Compare Data

if data=true

else

View Data

6) **Implementation**

We execute our program with the help of string and i/o libraries. To implement the goals following methodology are used.

**Step 1:** In this we start with a login page in which there will be various divisions like student information, fee details, room allotment, reports and others.

**Step 2:** In each division there will be further sub- divisions like in student info there will be name, gender, session, course and contact information and so on.

**Step 3:** We are using the concept of file handling for storing, reading and writing in database.

**Step4:**We are using bubble sort and binary search algorithm to make the program more efficient.

6.1) **Pseudocode**

Pseudocode for Bubble Sort

while(scan data entry){

qwe[k]=add;//qwe[k] is an array of data entered to be sorted

k++;

}

//sort in ascending oreder

for (i=k-1;i>=0;--i){

for(j=0;j<i;++j){

if(qwe[j].roll\_no>qwe[j+1].roll\_no)(

swap( qwe[j], qwe[j+1] )

}}

Pseudocode for Binary search

1. Let first = 0 and last = k-1.

2. Compute mid as the average of max and min, rounded down (so that it is an integer).

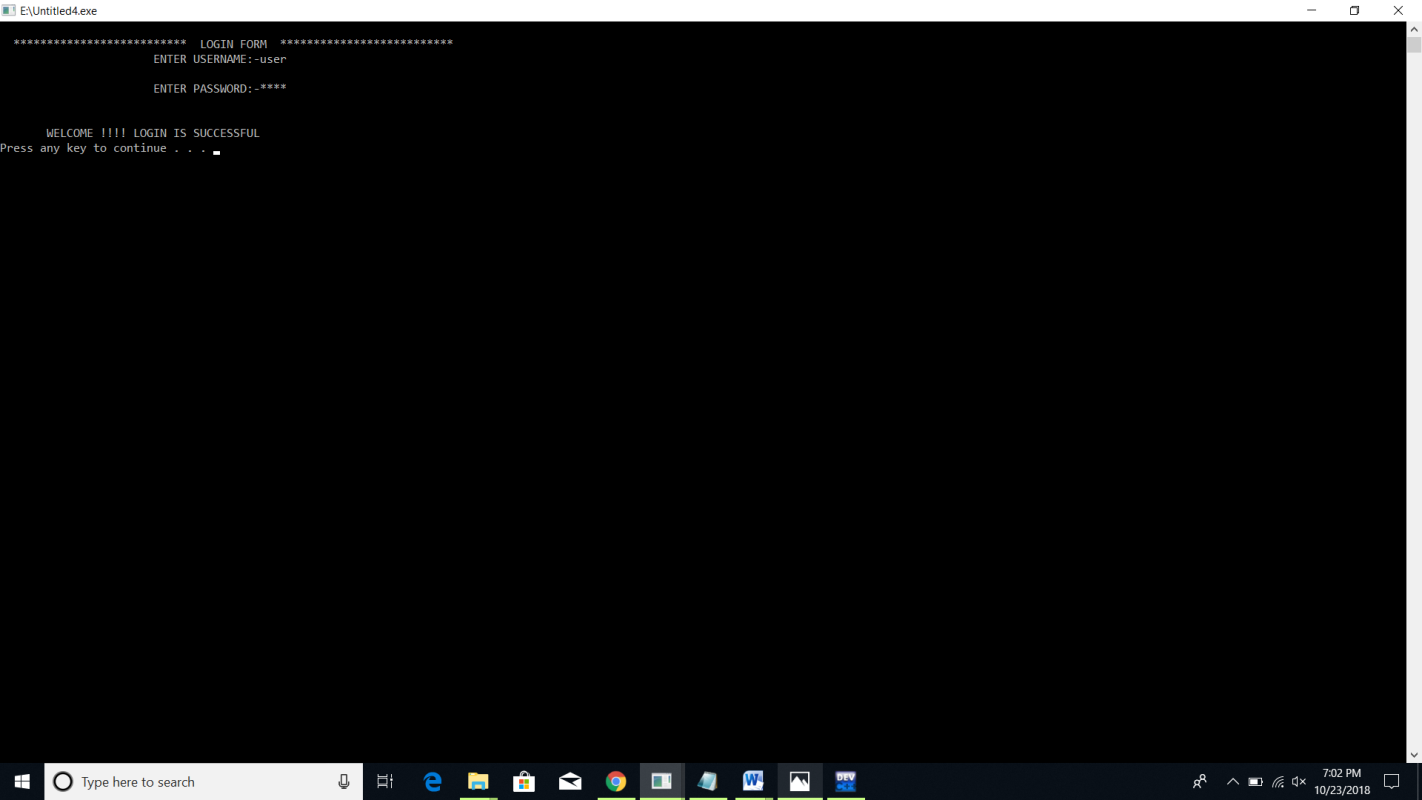
3. If array[mid] equals target, then stop. You found it! Return mid.

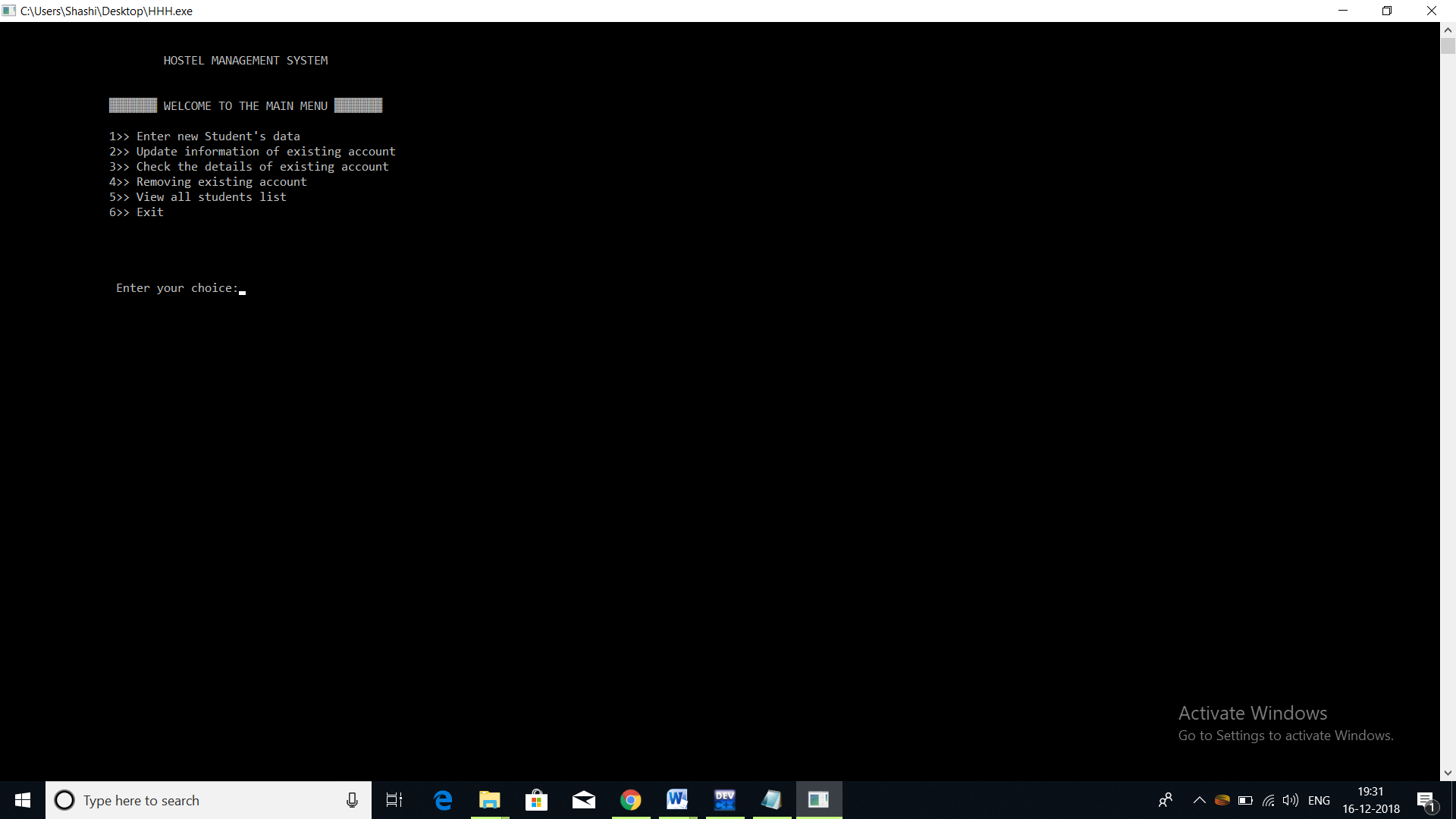
4. If the mid was too low, that is, array[mid] < target, then set first = mid + 1.

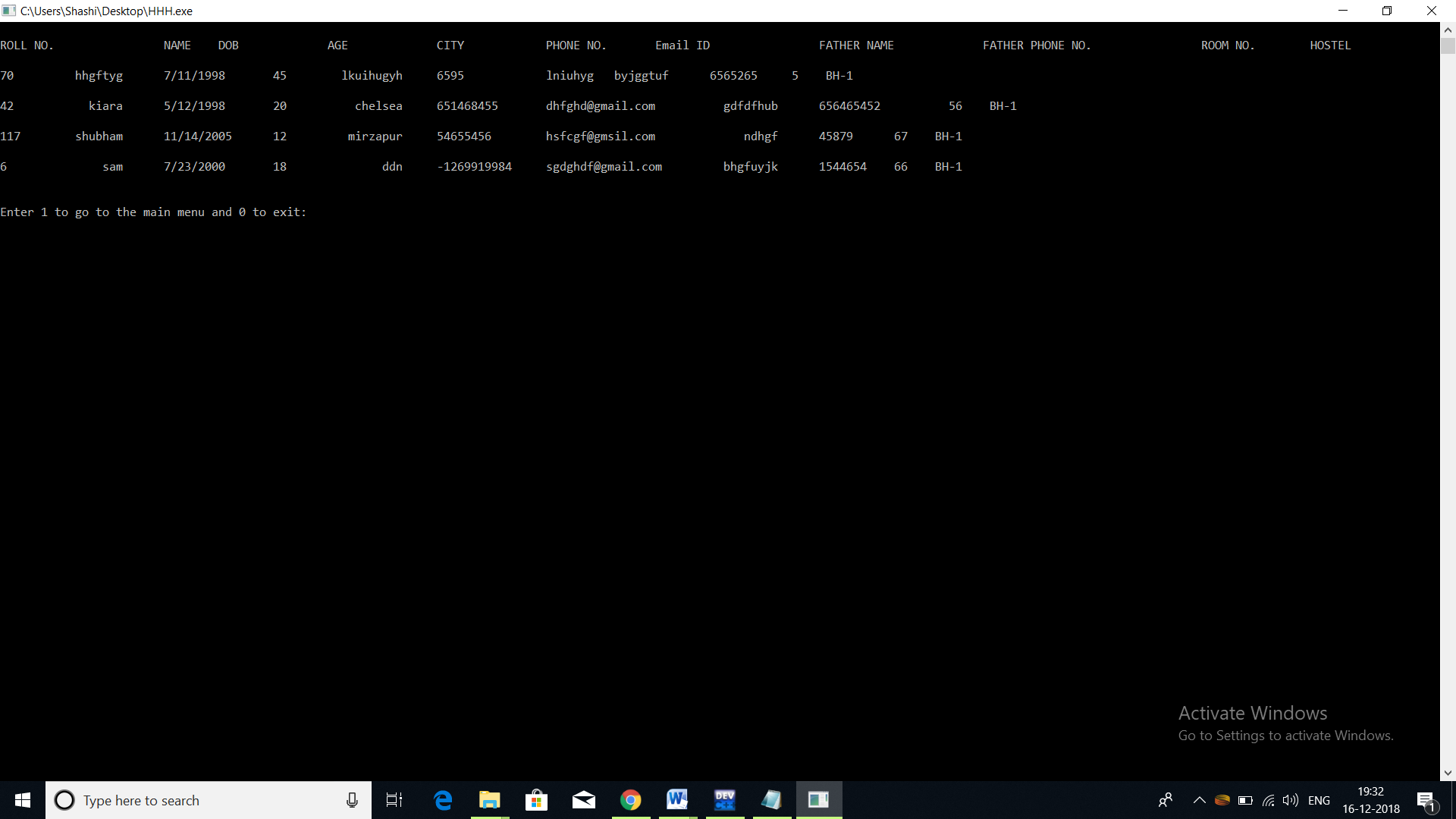
5. Otherwise, the mid was too high. Set last = mid - 1.

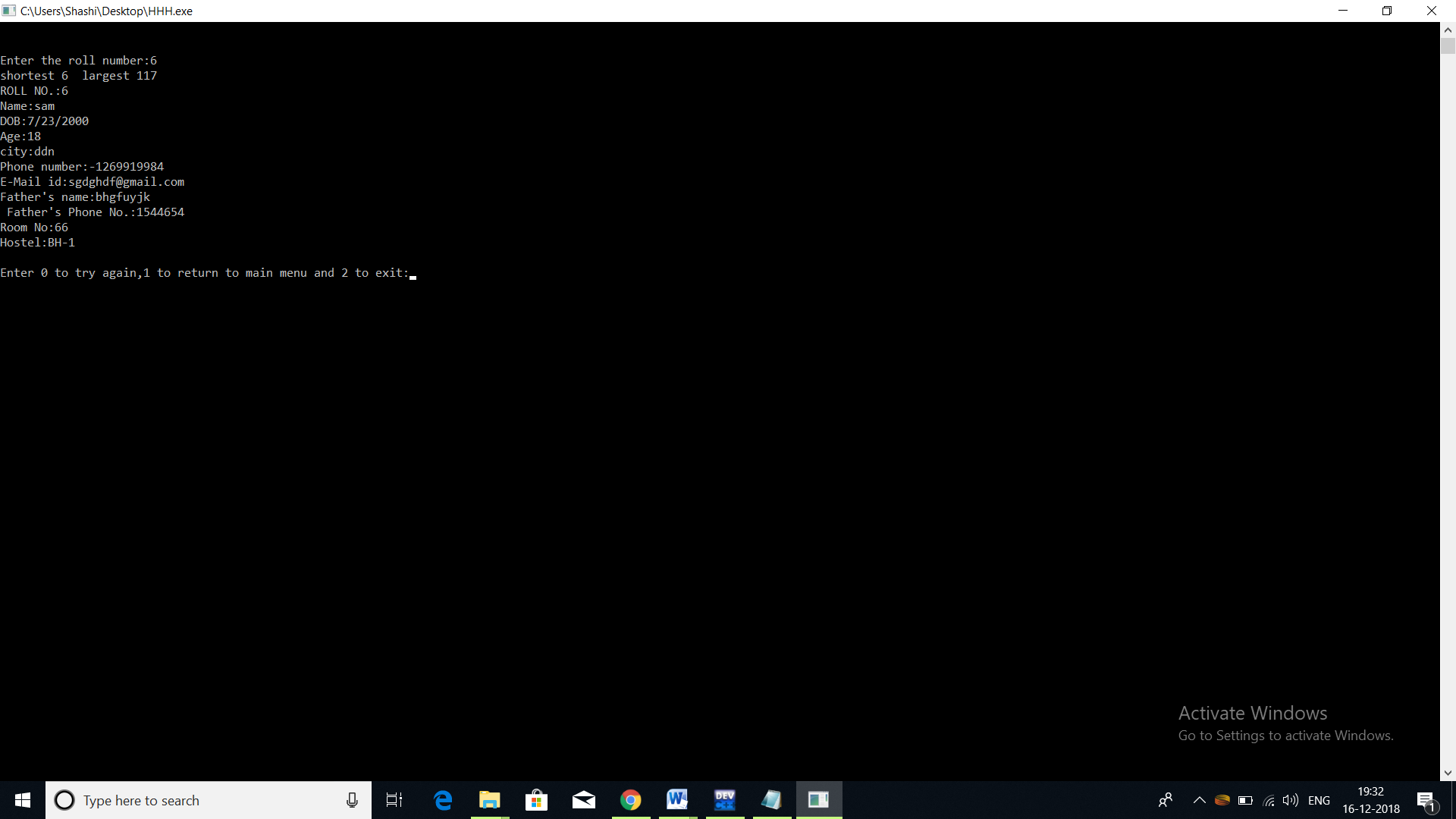
6. Go back to step 2.

6.2) **Output Screen**









7) References:-

[1]<https://www.slideshare.net/PrasoonRawat/hostel-management-system-report> Accessed on 4th Sep 2018.

[2] Accessed on 9th sep.

<http://www.freestudentprojects.com/studentprojectreport/projectreport/hostel-management-system-3/> Accessed on 9th Oct 2018.

[3]<http://dkingsnet.blogspot.com/2013/04/abstract-and-introduction-for-hostel.html> Accessed on 20th Oct 2018.

[4]<https://www.google.co.in/search?q=hardware+requirements+for+hostel+management+system&rlz=1C1CHBD_enIN798IN798&oq=system+requirements+for+hostel&aqs=chrome.1.69i57j0l4.16028j0j7&sourceid=chrome&ie=UTF-8> Accessed on 15th Nov 2018.

A APPENDIXIPROJECTCODE

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

int i,j,main\_exit;

void menu();

void new\_entry();

void edit();

void view\_list();

void erase();

void see();

void sorter();

void closer();

struct date{

int month,day,year;

};

struct{

char name[60];

int age;

char city[60];

char mail\_id[30];

char room\_no[8];

char father\_name[60];

int father\_phone;

int phone;

char hostel\_name[6];

int roll\_no;

struct date dob;

}add,upd,rem,sort,qwe[100],temp;

int check;

int main()

{

pass();

menu();

return 0;

}

int pass(void)

{

int a=0,i=0;

char uname[10],c=' ';

char pword[10],code[10];

char user[10]="user";

char pass[10]="pass";

do

{

system("cls");

printf("\n \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* LOGIN FORM \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* ");

printf(" \n ENTER USERNAME:-");

scanf("%s", &uname);

printf(" \n ENTER PASSWORD:-");

while(i<10)

{

pword[i]=getch();

c=pword[i];

if(c==13) break;

else printf("\*");

i++;

}

pword[i]='\0';

i=0;

if(strcmp(uname,user)==0 && strcmp(pword,pass)==0)

{

printf(" \n\n\n WELCOME !!!! LOGIN IS SUCCESSFUL\n");

system("PAUSE");

break;

}

else

{

printf("\n SORRY !!!! LOGIN IS UNSUCESSFUL\n");

system("PAUSE");

a++;

getch();

}

}

while(a<=2);

if (a>2)

{

printf("\nSorry you have entered the wrong username and password for four times!!!");

getch();

}

system("cls");

}

void menu()

{

system("CLS");

int choice;

printf("\n\n\t\t\tHOSTEL MANAGEMENT SYSTEM");

printf("\n\n\n\t\t\xB2\xB2\xB2\xB2\xB2\xB2\xB2 WELCOME TO THE MAIN MENU \xB2\xB2\xB2\xB2\xB2\xB2\xB2");

printf("\n\n\t\t1>> Enter new Student's data\n\t\t2>> Update information of existing account\n\t\t3>> Check the details of existing account\n\t\t4>> Removing existing account\n\t\t5>> View all students list\n\t\t6>> Exit\n\n\n\n\n\t\t Enter your choice:");

scanf("%d",&choice);

switch(choice)

{

case 1:new\_entry();

break;

case 2:edit();

break;

case 3:see();

break;

case 4:erase();

break;

case 5:view\_list();

break;

case 6:closer();

break;

default:menu();

}

}

void new\_entry()

{system("CLS");

int choice;

FILE \*ptr,\*hostel;

ptr=fopen("record.dat","a+");

roll\_no:

printf("\t\t\t\xB2\xB2\xB2 ADD STUDENTS DATA \xB2\xB2\xB2\xB2");

printf("\nEnter your roll number:");

scanf("%d",&check);

while(fscanf(ptr,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",&add.roll\_no, &add.name, &add.dob.month, &add.dob.day, &add.dob.year, &add.age, add.city, &add.phone, &add.mail\_id, &add.father\_name, &add.father\_phone, &add.room\_no, &add.hostel\_name)!=EOF)

{

if (check==add.roll\_no)

{printf("Students roll no. already in use!");

goto roll\_no;

}

}

add.roll\_no=check;

printf("\nEnter the name:");

scanf("%s",&add.name);

int z=1;

while(z!=0){

int mm,dd,y;

printf("\nEnter the date of birth(mm/dd/yyyy):");

scanf("%d/%d/%d",&mm,&dd,&y);

if(y>=1900 && y<=9999)

{

if(mm>=1 && mm<=12)

{

if(((dd>=1 && dd<=31) && (mm==1 || mm==3 || mm==5 || mm==7 || mm==8 || mm==10 || mm==12)) || ((dd>=1 && dd<=30)&&(mm==4 || mm==6 || mm==9 || mm==11)) || ((dd>=1 && dd<=28) && (mm==2)) || (dd==29 && mm==2 && (y%400==0 || (y%4==0 && y%100!=0))))

{

add.dob.month=mm;

add.dob.year=y;

add.dob.day=dd;

}

else

{printf("Day is invalid.\n");

continue;

}

}

else

{printf("Month is not valid.\n");

continue;}

}

else

{

printf("Year is not valid.\n");

continue;

}

z=0;

}

printf("\nEnter the age:");

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

printf("\nEnter the city:");

scanf("%s",&add.city);

printf("\nEnter the phone number: ");

scanf("%d",&add.phone);

printf("\n Enter your email id :");

scanf("%s",&add.mail\_id);

printf("\n Enter your Father's name:");

scanf("%s",&add.father\_name);

printf("\n Enter your Father's phone no.:");

scanf("%d",&add.father\_phone);

printf("\n Enter your room no.:");

scanf("%s",&add.room\_no);

printf("\n Enter your hostel name from BH-1, BH-2, BH-3, GH-1 :");

scanf("%s",&add.hostel\_name);

fprintf(ptr,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",add.roll\_no, add.name, add.dob.month, add.dob.day, add.dob.year, add.age, add.city, add.phone, add.mail\_id, add.father\_name, add.father\_phone, add.room\_no, add.hostel\_name);

fclose(ptr);

printf("\nStudent added successfully!");

add\_invalid:

printf("\n\n\n\t\tEnter 1 to go to the main menu and 0 to exit:");

scanf("%d",&main\_exit);

if (main\_exit==1)

menu();

else if(main\_exit==0)

closer();

else

{

printf("\nInvalid!\a");

goto add\_invalid;

}

}

void edit()

{

system("CLS");

int choice,test=0;

FILE \*old,\*newrec;

old=fopen("record.dat","r");

newrec=fopen("new.dat","w");

printf("\nEnter the roll no. of the student whose info you want to change:");

scanf("%s",upd.roll\_no);

while(fscanf(old,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",&add.roll\_no, &add.name, &add.dob.month, &add.dob.day, &add.dob.year, &add.age, &add.city, &add.phone, &add.mail\_id, &add.father\_name, &add.father\_phone, &add.room\_no, &add.hostel\_name)!=EOF)

{

if (add.roll\_no==upd.roll\_no)

{ test=1;

printf("\nWhich information do you want to change?\n1.Room no.\n2.Phone\n\nEnter your choice(1 for room no. and 2 for phone ):");

scanf("%d",&choice);

if(choice==1)

{

printf("Enter the new room no.:");

scanf("%s",&upd.room\_no);

fprintf(newrec,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",add.roll\_no, add.name, add.dob.month, add.dob.day, add.dob.year, add.age, add.city, add.phone, add.mail\_id, add.father\_name, add.father\_phone, upd.room\_no, add.hostel\_name);

printf("Changes saved!");

}

else if(choice==2)

{

printf("Enter the new phone number:");

scanf("%d",&upd.phone);

fprintf(newrec,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",add.roll\_no, add.name, add.dob.month, add.dob.day, add.dob.year, add.age, add.city, upd.phone, add.mail\_id, add.father\_name, add.father\_phone, add.room\_no, add.hostel\_name);

printf("Changes saved!");

}

}

else

fprintf(newrec,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",add.roll\_no, add.name, add.dob.month, add.dob.day, add.dob.year, add.age, add.city, add.phone, add.mail\_id, add.father\_name, add.father\_phone, add.room\_no, add.hostel\_name);

}

fclose(old);

fclose(newrec);

remove("record.dat");

rename("new.dat","record.dat");

if(test!=1)

printf("\nRecord not found!!\a\a\a");

edit\_invalid:

printf("\nEnter 0 to try again,1 to return to main menu and 2 to exit:");

scanf("%d",&main\_exit);

if (main\_exit==1)

menu();

else if (main\_exit==2)

closer();

else if(main\_exit==0)

edit();

else

{

printf("\nInvalid!\a");

goto edit\_invalid;

}

}

void see()

{int i,j,k=0;

system("CLS");

FILE \*ptr;

int test=0;

ptr=fopen("record.dat","r");

printf("\n\nEnter the roll number:");

scanf("%d",&check);

while (fscanf(ptr,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",&add.roll\_no, &add.name, &add.dob.month, &add.dob.day, &add.dob.year, &add.age, &add.city, &add.phone, &add.mail\_id, &add.father\_name, &add.father\_phone, &add.room\_no, &add.hostel\_name)!=EOF)

{

qwe[k]=add;

k++;

}

for (i=k-1;i>=0;--i){

for(j=0;j<i;++j){

if(qwe[j].roll\_no>qwe[j+1].roll\_no){

temp=qwe[j];

qwe[j]=qwe[j+1];

qwe[j+1]=temp;

}

}

}

printf("shortest %d largest %d",qwe[0].roll\_no,qwe[k-1].roll\_no);

int first=0,last=k-1,mid=(first+last)/2;

while(first<=last){

if(qwe[mid].roll\_no < check){

first = mid+1;

}

else if(qwe[mid].roll\_no == check){

test=1;

printf("\nROLL NO.:%d\nName:%s \nDOB:%d/%d/%d \nAge:%d \ncity:%s \nPhone number:%d \nE-Mail id:%s \nFather's name:%s \n Father's Phone No.:%d \nRoom No:%s \nHostel:%s \n",add.roll\_no, add.name, add.dob.month, add.dob.day, add.dob.year, add.age, add.city, add.phone, add.mail\_id, add.father\_name, add.father\_phone, add.room\_no, add.hostel\_name);

break;

}

else

last = mid-1;

mid = (first+last)/2;

}

fclose(ptr);

if(test!=1)

printf("\nRecord not found!!\a\a\a");

see\_invalid:

printf("\nEnter 0 to try again,1 to return to main menu and 2 to exit:");

scanf("%d",&main\_exit);

if (main\_exit==1)

menu();

else if (main\_exit==2)

closer();

else if(main\_exit==0)

erase();

else

{

printf("\nInvalid!\a");

goto see\_invalid;

}

}

void erase()

{

system("CLS");

FILE \*old,\*newrec;

int test=0;

old=fopen("record.dat","r");

newrec=fopen("new.dat","w");

printf("Enter the roll no. of student whose data you want to delete:");

scanf("%d",&rem.roll\_no);

while (fscanf(old,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",&add.roll\_no, &add.name, &add.dob.month, &add.dob.day, &add.dob.year, &add.age, &add.city, &add.phone, &add.mail\_id, &add.father\_name, &add.father\_phone, &add.room\_no, &add.hostel\_name)!=EOF)

{

if(add.roll\_no!=rem.roll\_no)

fprintf(newrec,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",add.roll\_no, add.name, add.dob.month, add.dob.day, add.dob.year, add.age, add.city, add.phone, add.mail\_id, add.father\_name, add.father\_phone, add.room\_no, add.hostel\_name);

else

{test++;

printf("\nRecord deleted successfully!\n");

}

}

fclose(old);

fclose(newrec);

remove("record.dat");

rename("new.dat","record.dat");

if(test==0)

printf("\nRecord not found!!\a\a\a");

erase\_invalid:

printf("\nEnter 0 to try again,1 to return to main menu and 2 to exit:");

scanf("%d",&main\_exit);

if (main\_exit==1)

menu();

else if (main\_exit==2)

closer();

else if(main\_exit==0)

erase();

else

{

printf("\nInvalid!\a");

goto erase\_invalid;

}

}

void view\_list()

{

system("CLS");

FILE \*view;

view=fopen("record.dat","r");

int test=0;

printf("\nROLL NO.\t\tNAME\tDOB\t\tAGE\t\tCITY\t\tPHONE NO.\tEmail ID\t\tFATHER NAME\t\tFATHER PHONE NO.\t\tROOM NO.\tHOSTEL\n");

while(fscanf(view,"%d %s %d/%d/%d %d %s %d %s %s %d %s %s\n",&add.roll\_no, &add.name, &add.dob.month, &add.dob.day, &add.dob.year, &add.age, &add.city, &add.phone, &add.mail\_id, &add.father\_name, &add.father\_phone, &add.room\_no, &add.hostel\_name)!=EOF)

{

printf("\n%d\t%10s\t%d/%d/%d\t%d\t %10s\t%d\t%15s\t%10s\t%d\t%5s\t%5s\n",add.roll\_no,add.name,add.dob.month,add.dob.day,add.dob.year,add.age,add.city,add.phone,add.mail\_id,add.father\_name,add.father\_phone,add.room\_no,add.hostel\_name);

test++;

}

fclose(view);

if (test==0)

{

printf("\nNO RECORDS!!\n");

}

view\_list\_invalid:

printf("\n\nEnter 1 to go to the main menu and 0 to exit:");

scanf("%d",&main\_exit);

if (main\_exit==1)

menu();

else if(main\_exit==0)

closer();

else

{

printf("\nInvalid!\a");

goto view\_list\_invalid;

}

}

void closer()

{

system("CLS");

printf("\n\nTHANK YOU!");

}

8