**DATA STRUCTURE AND ALGORITHM**

**------ PROJECT REPORT ------**

* **TEAM MEMBERS:**

1. **KARUNESH TRIPATHI – 19BCE0880**
2. **MAHIN JAIN – 19BDS0086**
3. **MANIK MAHESHWARI – 19BDS0129**
4. **JEETESH GOWDER – 19BCE2176**

**SLOT- L11+12**

**FACULTY- DR. PARVEEN SULTANA H**

* AIM & OBJECTIVE:

The aim of this project is to study, design and implement a hostel administration process using binary search tree data structure.

* ABSTRACT:

Our project deals with a Hostel Administration System which stores necessary details of the students such as name, address and other personal information required by the administration.

It give’s option to login as administrator and as student so the person who is using the software is able to use the different features of the software.

* BASIC PRINCIPLE AND METHODOLOGY:

Our C Program code is based on the records necessary for a hostel when a student has been admitted in it. The hostel administrative system also stores basic information of students for reference. It makes it easier for faculty and administrative members to access information of a particular student and know more about his/her details as well as allows student to login and update his or her details.

* PROCESS MODULE:



* CODE:

#include <stdio.h>

#include <stdlib.h>

#include<string.h>

typedef struct student

{

char regno[10];

char password[20];

char name[30];

char email[50];

char address[100];

char fathername[30];

int age;

long long int mobile;

long long int fathermobile;

int roomno;

char log[10000];

char block[2];

int secno;

struct student \*left;

struct student \*right;

}node;

node \*root=NULL;

void login()

{

int ch=0;

node \*troot=root;

int flag=0;

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER YOUR REGISTRATION NUMBER : ");

char rno[10];char pass[20];

scanf("%s",rno);

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER PASSWORD : ");

scanf("%s",pass);

while(troot!=NULL)

{

if(strcmp(rno,troot->regno)<0)

troot=troot->left;

else if((strcmp(rno,troot->regno)==0)&&(strcmp(pass,troot->password)==0))

{

flag=1;

break;

}

else

troot=troot->right;

}

if(flag==1)

{

printf("\n\n\n\t\t\t\t\t\t\t\t\t\t\tENTER: \n\n\t\t\t\t\t\t\t\t\t\t\t1. TO SEE YOUR DETAILS\n\t\t\t\t\t\t\t\t\t\t\t2. TO CHANGE EMAIL\n\t\t\t\t\t\t\t\t\t\t\t3. TO CHANGE ADDRESS\n\t\t\t\t\t\t\t\t\t\t\t4. TO CHANGE YOUR MOBILE NUMBER\n\t\t\t\t\t\t\t\t\t\t\t5. TO CHANGE FATHER'S MOBILE NUMBER\n\t\t\t\t\t\t\t\t\t\t\t6. TO CHANGE SECURITY NUMBER\n\n\t\t\t\t\t\t\t\t\t\t\tCHOICE: ");

scanf("%d",&ch);

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

switch(ch)

{

case 1:

{

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tREGISTRATION NUMBER: %s",troot->regno);

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tNAME: %s",troot->name);

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tAGE: %d",troot->age);

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tEMAIL: %s",troot->email);

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tADDRESS: %s",troot->address);

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tMOBILE NUMBER: %lld",troot->mobile);

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tBLOCK: %s",troot->block);

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tROOM NUMBER: %d",troot->roomno);

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tFATHER'S NAME: %s",troot->fathername);

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tFATHER'S MOBILE NUMBER: %lld\n",troot->fathermobile);

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

break;

}

case 2:

{

char s[50];

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER YOUR NEW EMAIL: ");

scanf("%s",s);

strcpy(troot->email,s);

printf("\n\t\t\t\t\t\t\t\t\t\t\tEMAIL SUCCESSFULLY UPDATED\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

break;

}

case 3:

{

getchar();

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER YOUR NEW ADDRESS: ");

scanf("%[^\n]s",troot->address);

printf("\n\t\t\t\t\t\t\t\t\t\t\tADDRESS SUCCESSFULLY UPDATED\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

break;

}

case 4:

{

long long int s;

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER YOUR NEW MOBILE NUMBER: ");

scanf("%lld",&s);

troot->mobile=s;

printf("\n\t\t\t\t\t\t\t\t\t\t\tMOBILE NUMBER SUCCESSFULLY UPDATED\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

break;

}

case 5:

{

long long int s;

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER YOUR NEW FATHER'S MOBILE NUMBER: ");

scanf("%lld",&s);

troot->fathermobile=s;

printf("\n\t\t\t\t\t\t\t\t\t\t\tFATHER'S MOBILE NUMBER SUCCESSFULLY UPDATED\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

break;

}

case 6:

{

int s;

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER YOUR NEW SECURITY NUMBER: ");

scanf("%d",&s);

troot->secno=s;

printf("\n\t\t\t\t\t\t\t\t\t\t\tSECURITY NUMBER SUCCESSFULLY UPDATED\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

break;

}

default:

{

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER CORRECT CHOICE\n\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

}

}

else

{

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER CORRECT REGISTRATION NUMBER OR PASSWORD\n\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

}

node \*create()

{

node \*temp;

temp=(node\*)malloc(sizeof(node));

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER REGISTERATION NUMBER: ");

scanf("%s",temp->regno);

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER STUDENT'S NAME: ");

scanf("%[^\n]s",temp->name);

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER STUDENT'S AGE: ");

scanf("%d",&(temp->age));

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER PASSWORD THAT STUDENT WANT TO SET: ");

scanf("%[^\n]s",temp->password);

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER SECURITY NUMBER YOU WANT TO SET: ");

scanf("%d",&temp->secno);

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER STUDENT'S EMAIL ID: ");

scanf("%[^\n]s",temp->email);

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER STUDENT'S ADDRESS: ");

scanf("%[^\n]s",temp->address);

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER STUDENT'S MOBILE NUMBER: ");

scanf("%lld",&(temp->mobile));

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER STUDENT'S FATHER'S NAME: ");

scanf("%[^\n]s",temp->fathername);

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER STUDENT'S FATHER'S MOBILE NUMBER: ");

scanf("%lld",&(temp->fathermobile));

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER BLOCK ALLOTED THE STUDENT: ");

scanf("%s",(temp->block));

getchar();

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER ROOM NUMBER ALLOTED TO STUDENT: ");

scanf("%d",&(temp->roomno));

getchar();

temp->left=temp->right=NULL;

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

return(temp);

}

void insert(node \*r,node \*temp)

{

if(root==NULL)

{

root=temp;

}

else

{

if(strcmp((r->regno),(temp->regno))<0)

{

if(r->right!=NULL)

{

insert(r->right,temp);

}

else

{

r->right=temp;

}

}

else

{

if(r->left!=NULL)

{

insert(r->left,temp);

}

else

{

r->left=temp;

}

}

}

}

node \*findmax(node \*l)

{

node \*r=NULL;

char \*m="00aaa0000";

while(l!=NULL)

{

if(strcmp(l->regno,m)>0)

{

r=l;

m=l->regno;

l=l->right;

}

}

return r;

}

void removest()

{

char rn[10];

int pos=0;

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER THE REGISTRATION NUMBER OF THE STUDENT TO BE REMOVED : ");

scanf("%s",rn);

getchar();

node \*troot=root;

node \*prev=root;

while(troot!=NULL)

{

if(strcmp(rn,troot->regno)==0)

{

if(troot->left==NULL&&troot->right==NULL)

{

if(strcmp(root->regno,rn)==0)

{

printf("\t\t\t\t\t\t\t\t\t\t\tNO STUDENT LEFT\n");

}

free(troot);

}

else if((troot->left==NULL&&troot->right!=NULL)||(troot->right==NULL&&troot->left!=NULL))

{

if(troot->left==NULL&&pos==1)

{

prev->right=troot->right;

}

else if(troot->right==NULL&&pos==1)

{

prev->right=troot->left;

}

else if(troot->left==NULL&&pos==2)

{

prev->left=troot->right;

}

else if(troot->right==NULL&&pos==2)

{

prev->left=troot->left;

}

free(troot);

}

else

{

if(pos==1)

{

prev->right=troot->left;

node \*p=findmax(troot->left);

p->right=troot->right;

free(troot);

}

if(pos==2)

{

prev->left=troot->left;

node \*p=findmax(troot->left);

p->right=troot->right;

free(troot);

}

if(pos==0)

{

prev=prev->left;

troot=troot->right;

free(root);

root=troot;

while(troot->left!=NULL)

{

troot=troot->left;

printf("%s",troot->regno);

}

troot->left=prev;

}

}

}

else if(strcmp(rn,troot->regno)>0)

{prev=troot;troot=troot->right;pos=1;}

else

{prev=troot;troot=troot->left;pos=2;}

}

printf("\n\t\t\t\t\t\t\t\t\t\t\tSTUDENT SUCCESSFULLY REMOVED\n\n");

}

int main()

{

int ch=0,ich=0,flag=0,sn;char rn[10];int id=0;

for(int i=0;i<10;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

printf("\n\t\t\t\t\t\t\t\t\t\t VIT HOSTEL MANAGEMENT SYSTEM\n");

for(int i=0;i<10;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

printf("\n\n\n\t\t\t\t\t\t\tENTER 4 DIGIT ONE TIME ADMINISTRATOR PIN TO OPEN THE HOSTEL AS ADMINISTRATOR: ");

scanf("%d",&id);

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tPRESS ENTER TO CONTINUE ");

getchar();

getchar();

system("cls");

do

{

flag=0;

for(int i=0;i<10;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

printf("\n\t\t\t\t\t\t\t\t\t\t VIT HOSTEL MANAGEMENT SYSTEM\n");

for(int i=0;i<10;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

printf("\n\n\n\t\t\t\t\t\t\t\t\t\t\tENTER: \n\n\t\t\t\t\t\t\t\t\t\t\t1. FOR STUDENT LOGIN\n\n\t\t\t\t\t\t\t\t\t\t\t2.FOR ADMINISTRATIVE LOGIN\n\n\t\t\t\t\t\t\t\t\t\t\tCHOICE: ");

scanf("%d",&ch);

system("cls");

switch(ch)

{

case 1:

{

for(int i=0;i<10;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

printf("\t\t\t\t\t\t\t\t\t\t\t\t VIT HOSTEL MANAGEMENT SYSTEM\n");

for(int i=0;i<10;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

printf("\n\n\n\t\t\t\t\t\t\t\t\t\t\tENTER: \n\n\t\t\t\t\t\t\t\t\t\t\t1. TO LOGIN\n\t\t\t\t\t\t\t\t\t\t\t2. CHANGE PASSWORD\n\t\t\t\t\t\t\t\t\t\t\t3. TO SEE YOUR LOG\n\t\t\t\t\t\t\t\t\t\t\t4. TO GO BACK\n\n\t\t\t\t\t\t\t\t\t\t\tCHOICE: ");

scanf("%d",&ich);

if(ich==1)

{

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

login();

}

else if(ich==2)

{

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

node \*temp;

temp=root;

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER YOUR REGNO: ");

scanf("%s",rn);

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER YOUR SECURITY NUMBER: ");

scanf("%d",&sn);

while(temp!=NULL&&flag==0)

{

if(strcmp(rn,temp->regno)==0&&(sn==temp->secno))

{

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER NEW PASSWORD: ");

scanf("%s",temp->password);

printf("\n\t\t\t\t\t\t\t\t\t\t\tPASSWORD SUCCESSFULLY UPDATED\n\n");

flag=1;

}

else if(strcmp(rn,temp->regno)<0)

temp=temp->left;

else

temp=temp->right;

}

if(flag==0)

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER CORRECT REGISTRATION NUMBER OR SECURITY NUMBER\n\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

else if(ich==3)

{

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

flag=0;

node \*temp;

temp=root;

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER THE REGISTRATION NUMBER: ");

scanf("%s",rn);

while(temp!=NULL&&flag==0)

{

if(strcmp(temp->regno,rn)==0)

{

printf("\n\t\t\t\t\t\t\t\t\t\t\t%s\n\n",temp->log);

flag=1;

}

else if(strcmp(rn,temp->regno)<0)

temp=temp->left;

else

temp=temp->right;

}

if(flag==0)

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER CORRECT REGISTRATION NUMBER\n\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

else if(ich==4)

{

system("cls");

continue;

}

else

{

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER CORRECT CHOICE\n\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

getchar();

break;

}

case 2:

{

for(int i=0;i<10;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

printf("\t\t\t\t\t\t\t\t\t\t\t\t VIT HOSTEL MANAGEMENT SYSTEM\n");

for(int i=0;i<10;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

int id1=0;

printf("\n\n\n\t\t\t\t\t\t\t\t\t\t\tENTER ADMINISTRATOR PIN: ");

scanf("%d",&id1);

getchar();

if(id==id1)

{

printf("\n\n\t\t\t\t\t\t\t\t\t\t\t ENTER:- \n\n\t\t\t\t\t\t\t\t\t\t\t1. TO ADD A NEW STUDENT\n\t\t\t\t\t\t\t\t\t\t\t2. TO REMOVE A STUDENT\n\t\t\t\t\t\t\t\t\t\t\t3. TO CHANGE ROOM NUMBER OF STUDENT\n\t\t\t\t\t\t\t\t\t\t\t4. TO ENTER DATA IN STUDENT'S LOG\n\t\t\t\t\t\t\t\t\t\t\t5. TO GO BACK\n\t\t\t\t\t\t\t\t\t\t\t6. TO SHUT DOWN THE HOSTEL\n\n\t\t\t\t\t\t\t\t\t\t\tCHOICE: ");

scanf("%d",&ich);

switch(ich)

{

case 1:

{

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

node \*temp=NULL;

temp=create();

insert(root,temp);

break;

}

case 2:

{

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

removest();

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

break;

}

case 3:

{

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

node \*temp;

temp=root;

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER THE REGISTRATION NUMBER OF STUDENT WHOSE ROOM HAS TO BE CHANGED: ");

scanf("%s",rn);

getchar();

while(temp!=NULL&&flag==0)

{

if(strcmp(rn,temp->regno)==0)

{

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER THE NEW ROOM NUMBER: ");

scanf("%d",&temp->roomno);

getchar();

flag=1;

}

else if(strcmp(rn,temp->regno)<0)

temp=temp->left;

else

temp=temp->right;

}

if(flag==0)

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER THE CORRECT REGISTRATION NUMBER\n");

printf("\n\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

break;

}

case 4:

{

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

node \*temp;

temp=root;

char l[1000];

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER REGISTRATION NUMBER OF STUDENT: ");

scanf("%s",rn);

getchar();

while(temp!=NULL&&flag==0)

{

if(strcmp(rn,temp->regno)==0)

{

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER DATA TO BE ENTERED: ");

scanf("%[^\n]s",l);

getchar();

strcat(l,"\n\t\t\t\t\t\t\t\t\t\t\t");

strcat(temp->log,l);

flag=1;

}

else if(strcmp(rn,temp->regno)<0)

temp=temp->left;

else

temp=temp->right;

}

if(flag==0)

printf("\n\t\t\t\t\t\t\t\t\t\t\tENTER THE CORRECT REGISTRATION NUMBER\n");

printf("\n\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

break;

}

case 5:

{

system("cls");

continue;

}

case 6:

{

exit("\n\n\t\t\t\t\t\t\t\t\t\t\tHOSTEL HAS BEEN SHUT DOWN\n\n");

}

default:

{

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER CORRECT CHOICE\n\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

}

}

else

{

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER CORRECT ADMINISTRATOR PIN\n");

for(int i=0;i<3;i++)

printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

break;

}

default:printf("\n\n\t\t\t\t\t\t\t\t\t\t\tENTER CORRECT CHOICE\n");

}

printf("\n\n\t\t\t\t\t\t\t\t\t\t\tPRESS ENTER TO CONTINUE ");

getchar();

system("cls");

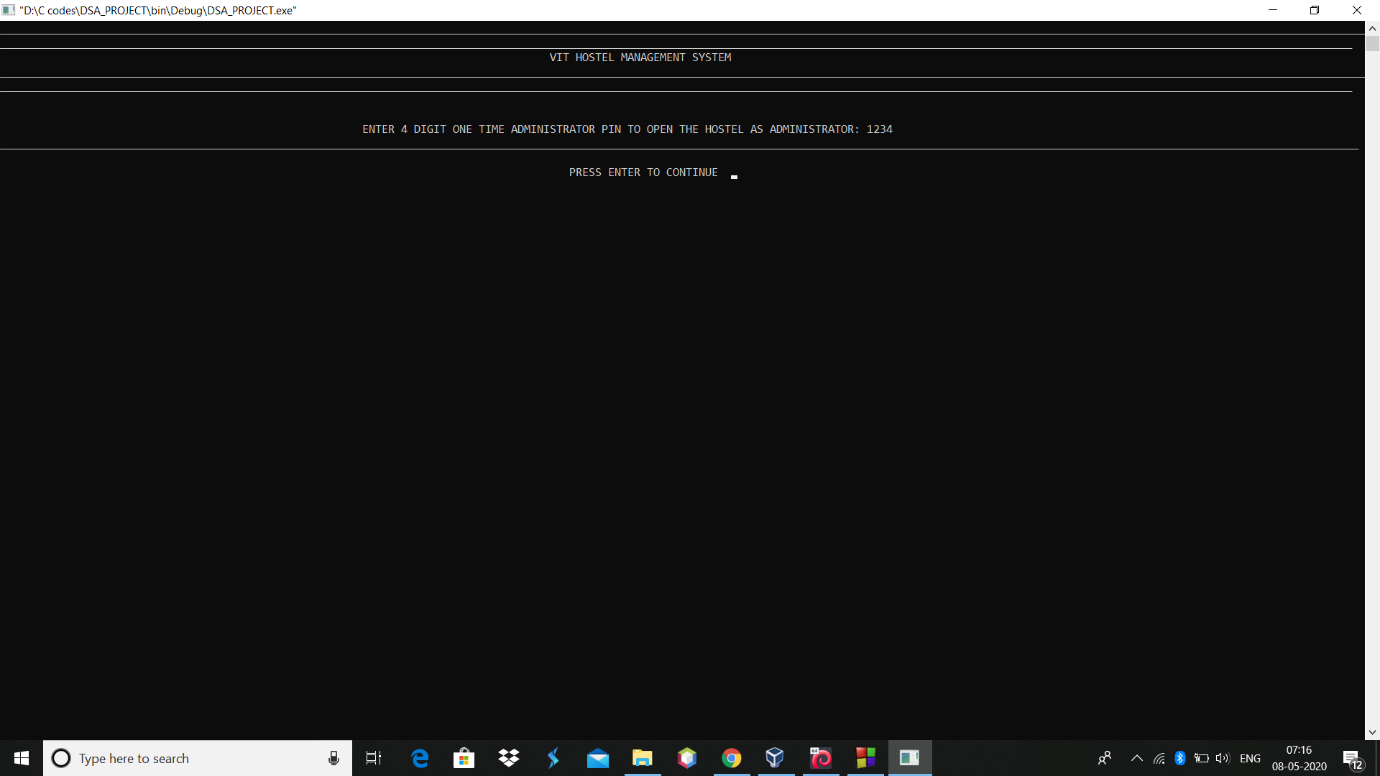
}while(5>1);

return 0;

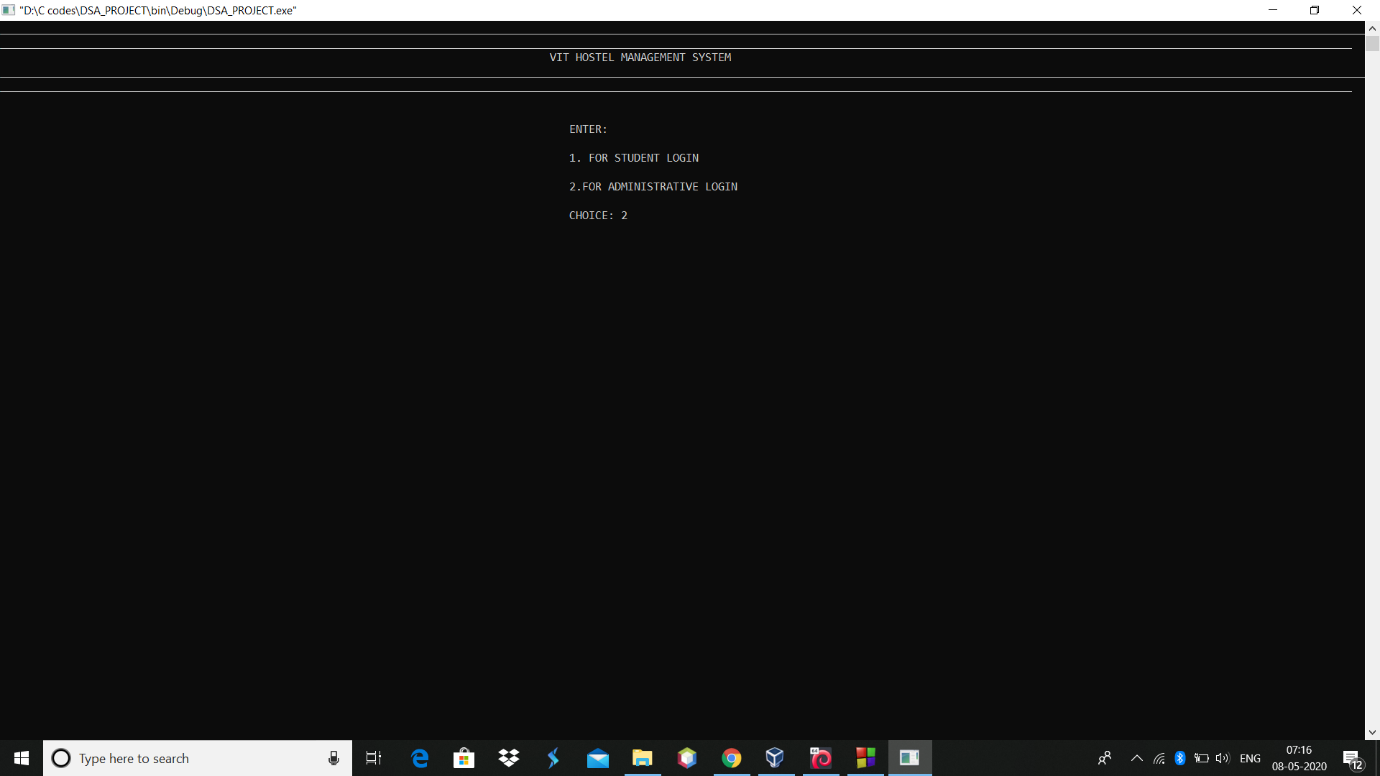
}

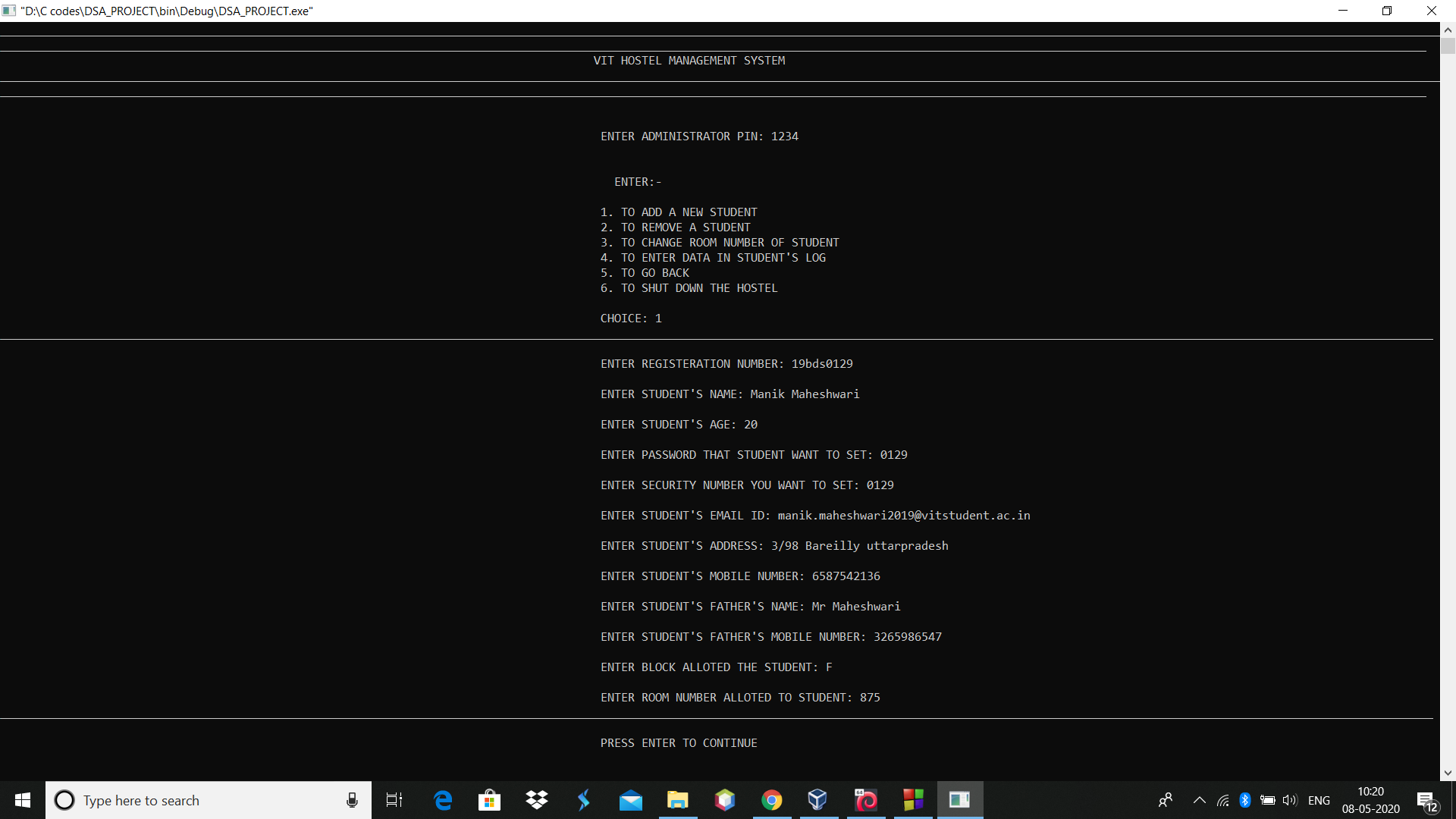
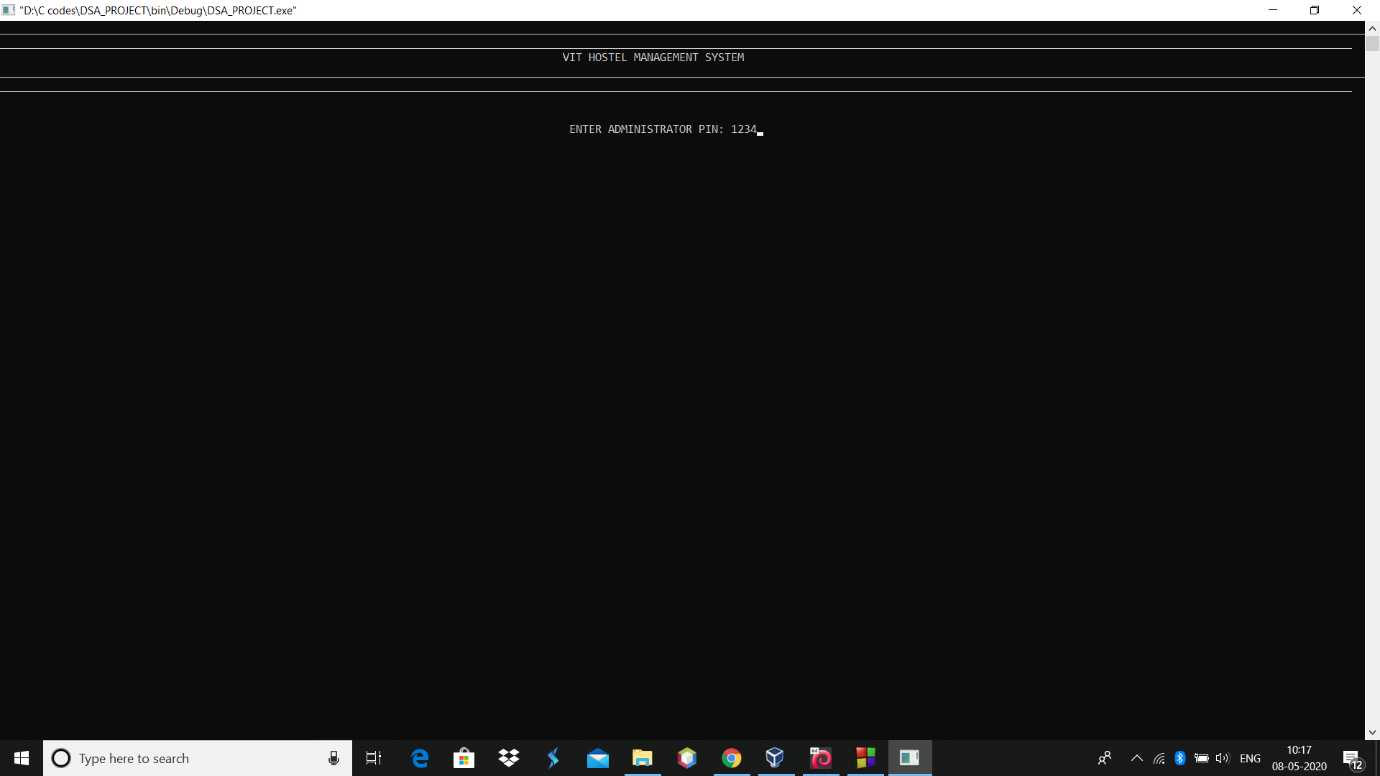
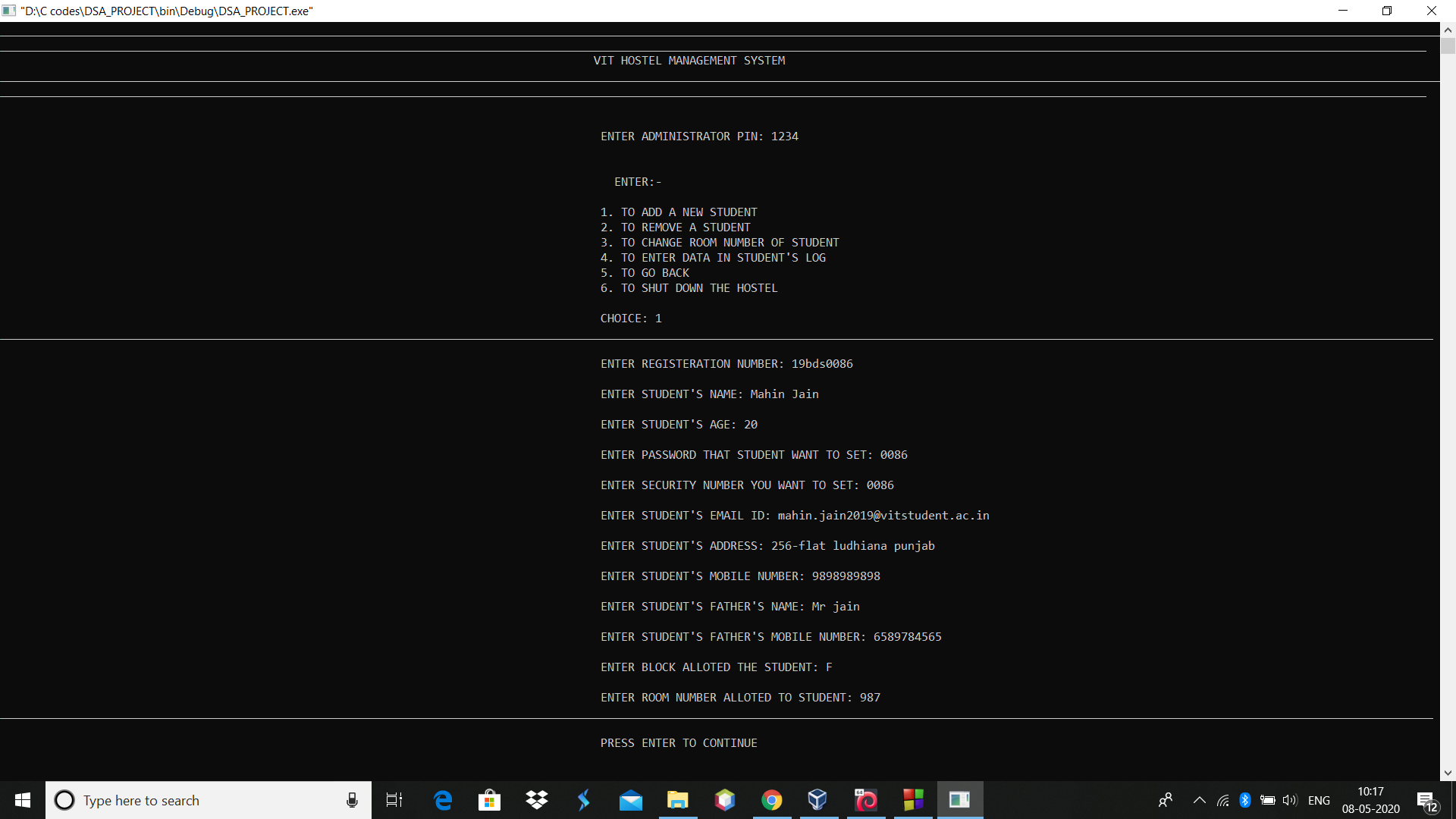
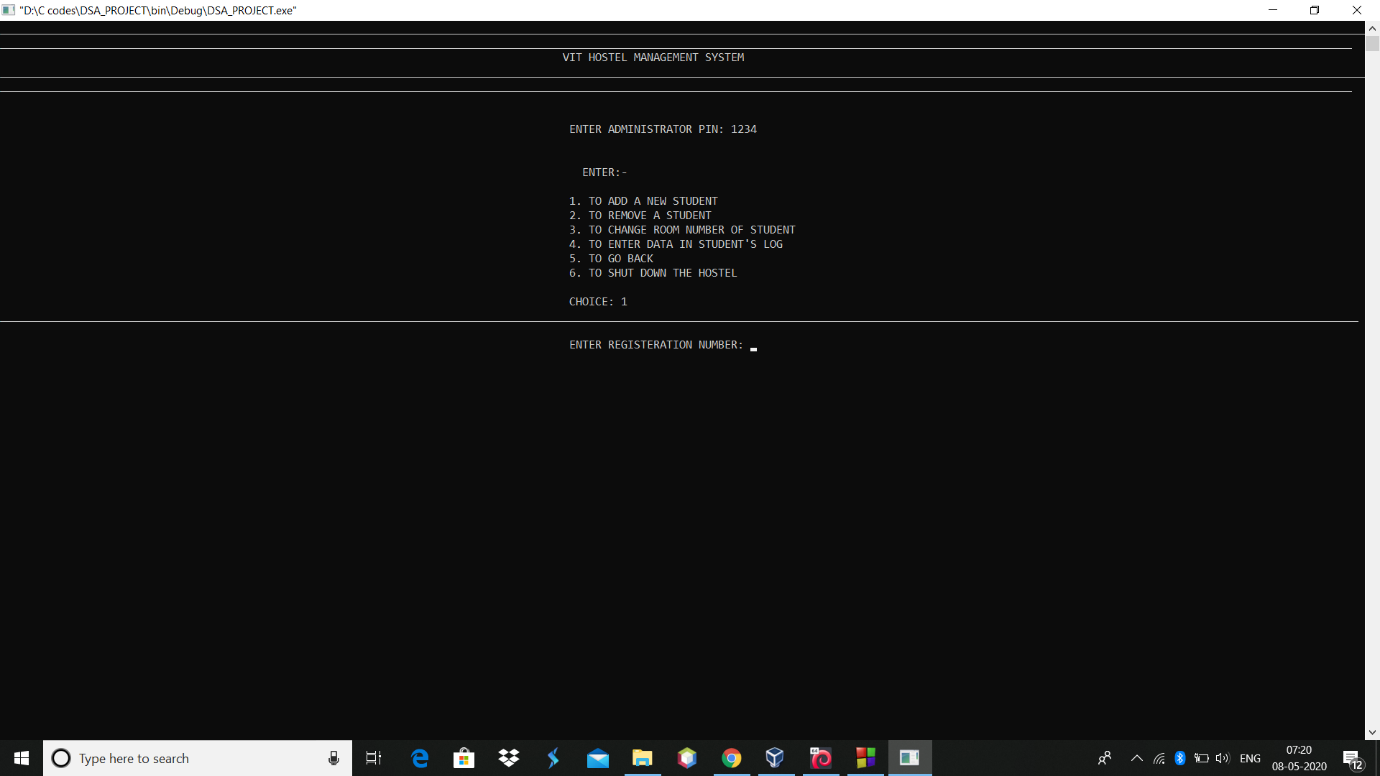
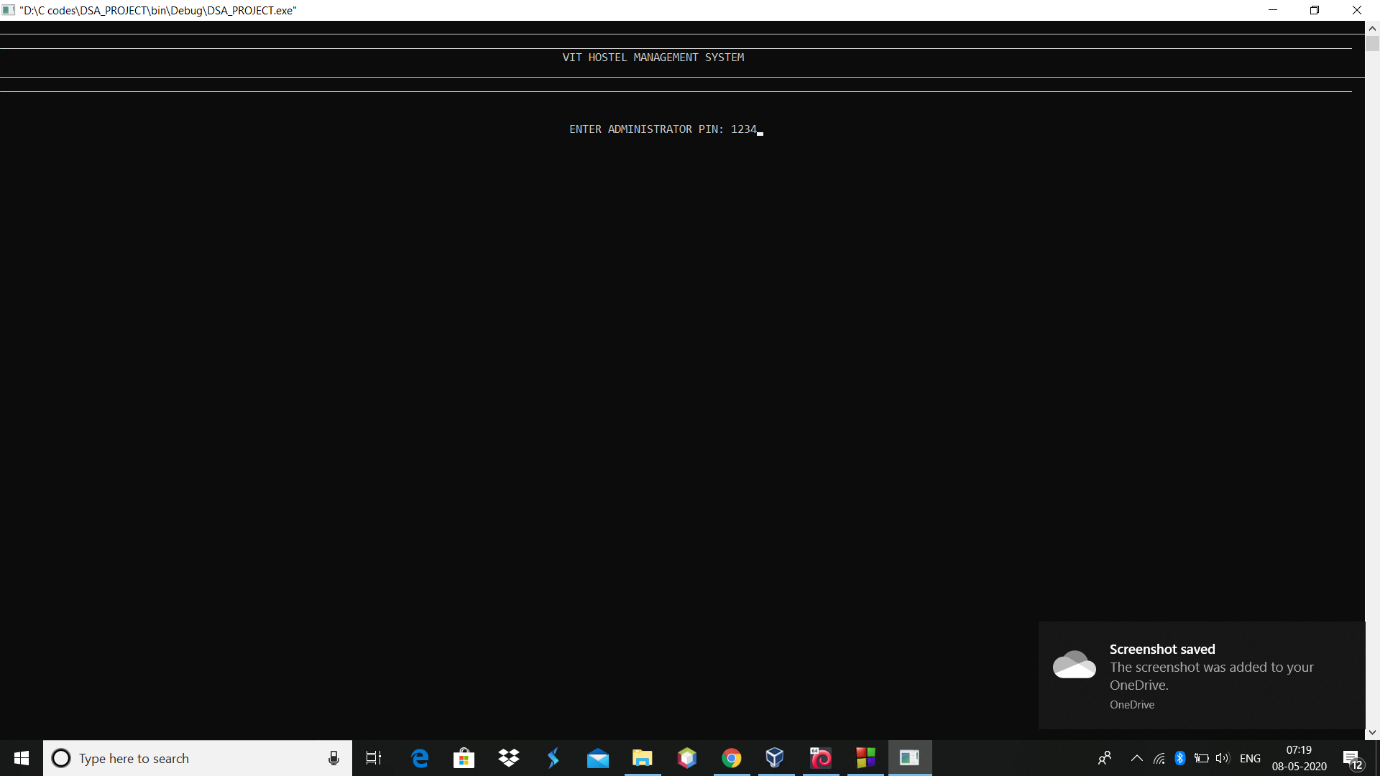
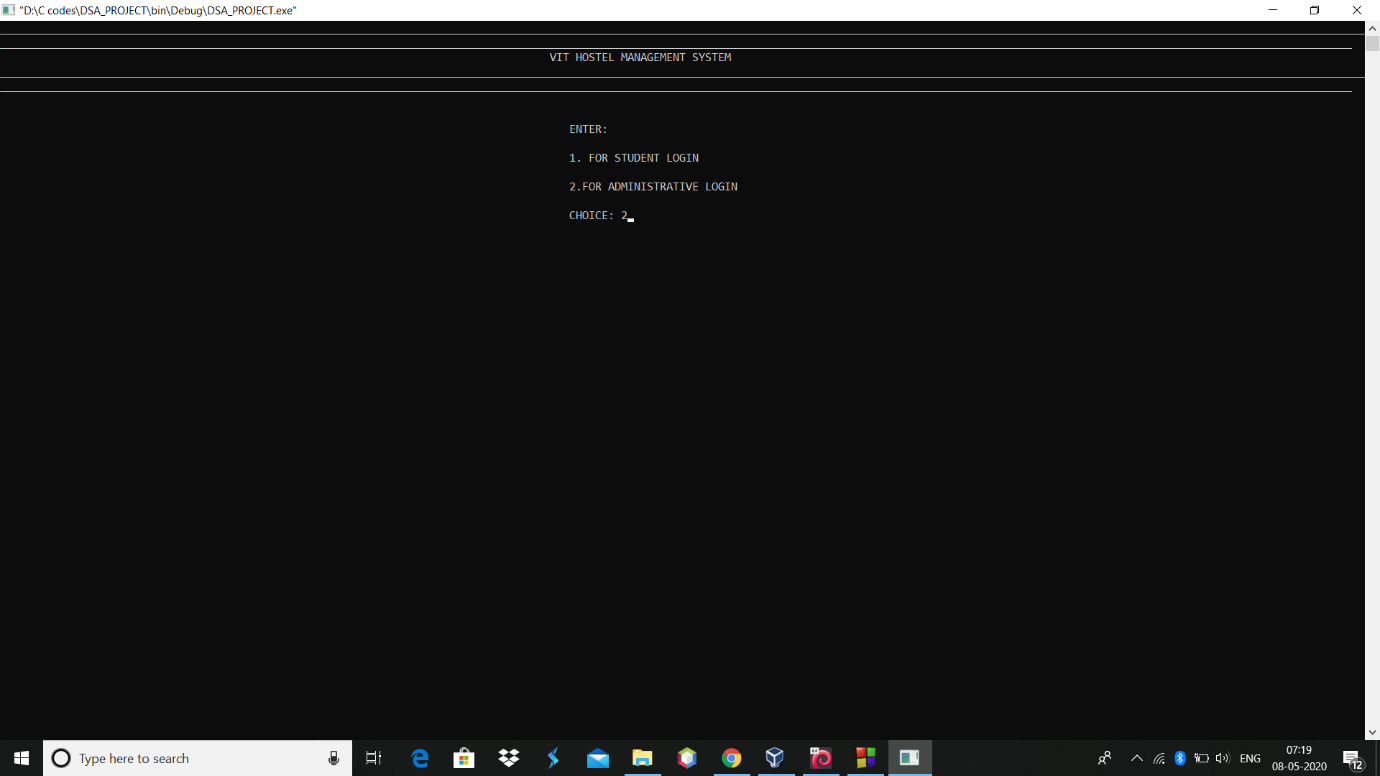
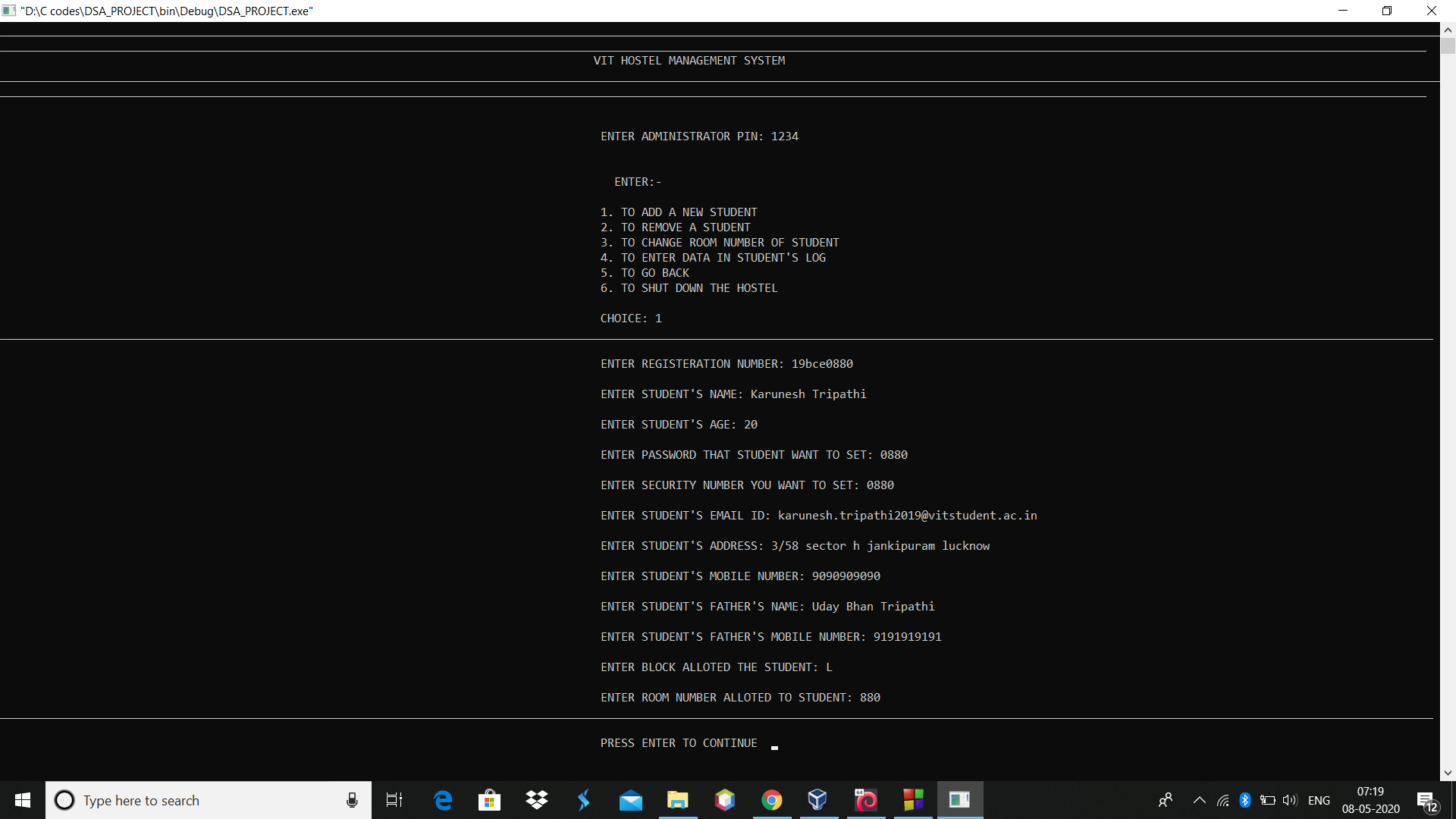
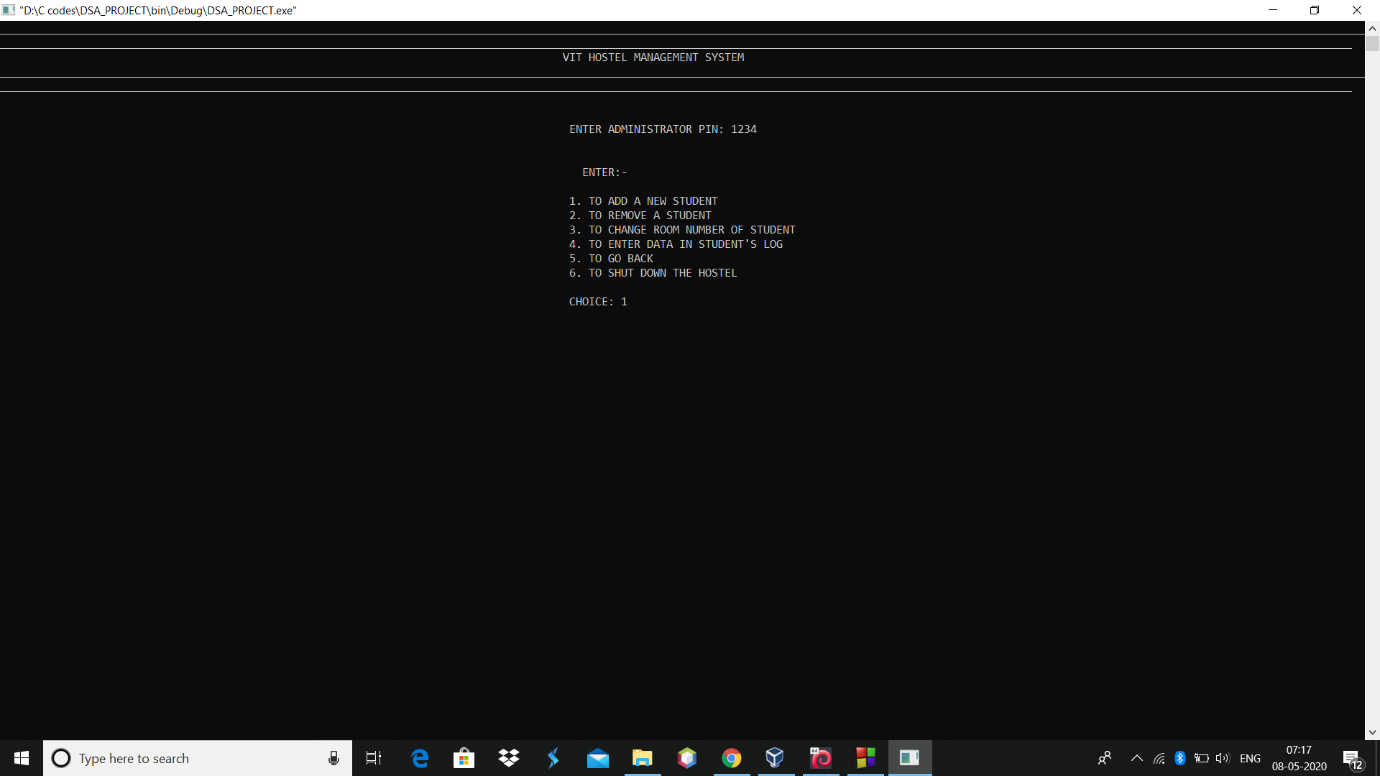
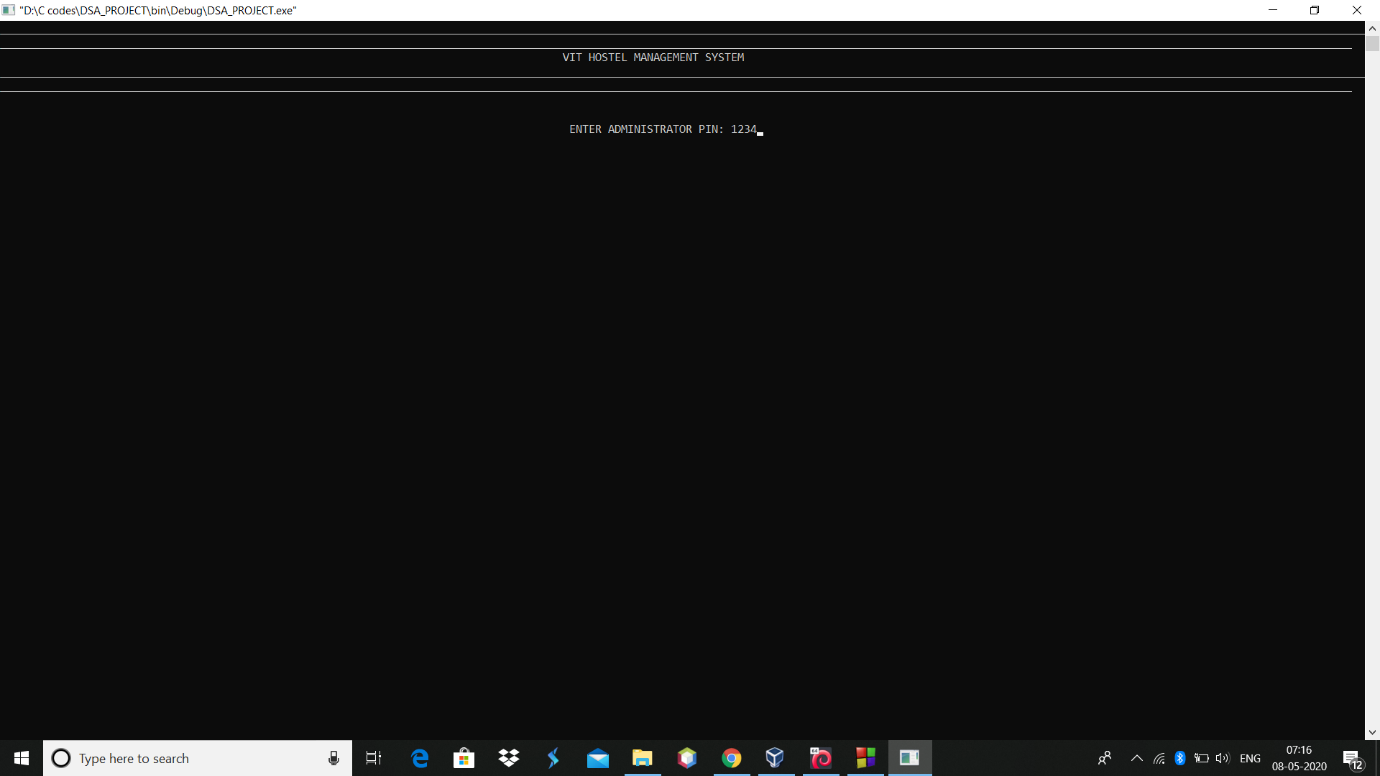
* OUTPUT SCREEN SHOT:-

**TO OPEN THE HOSTEL FOR 1ST TIME**

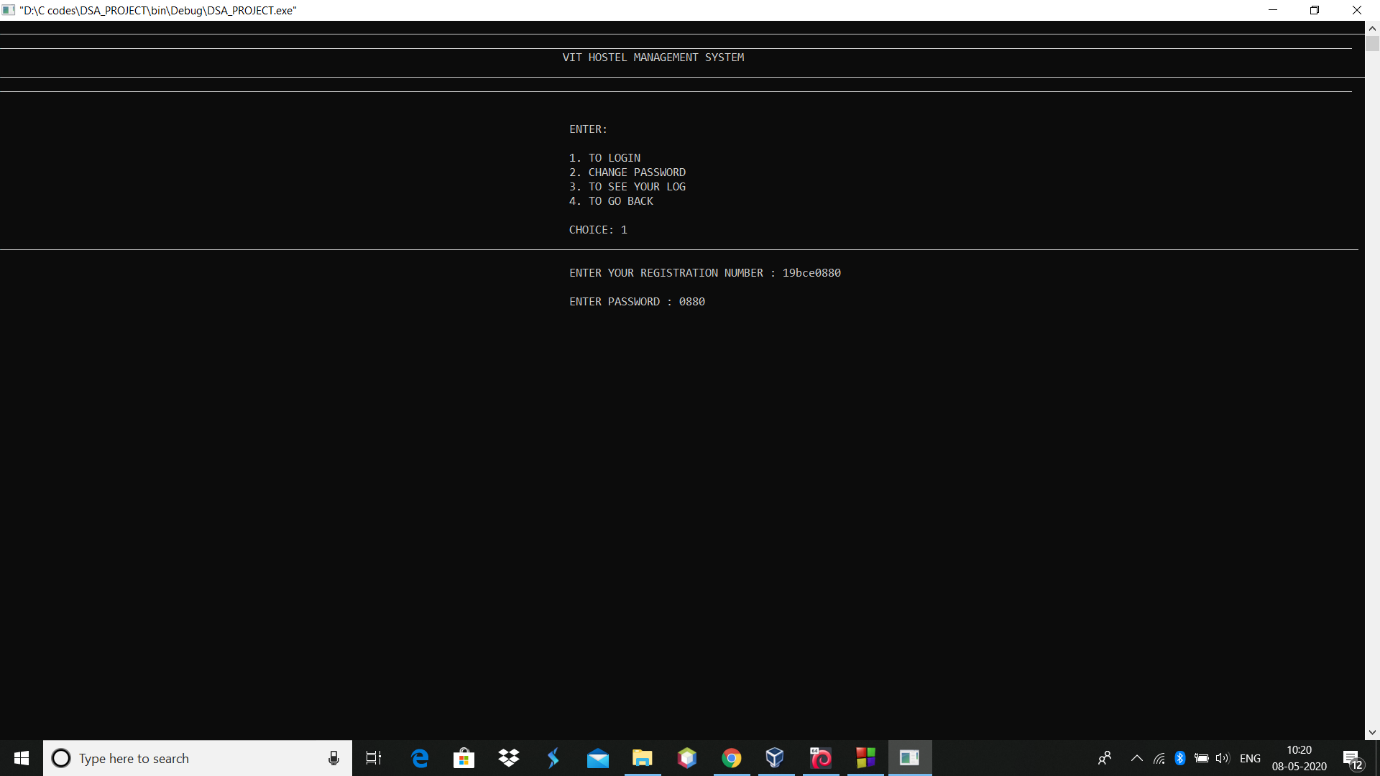
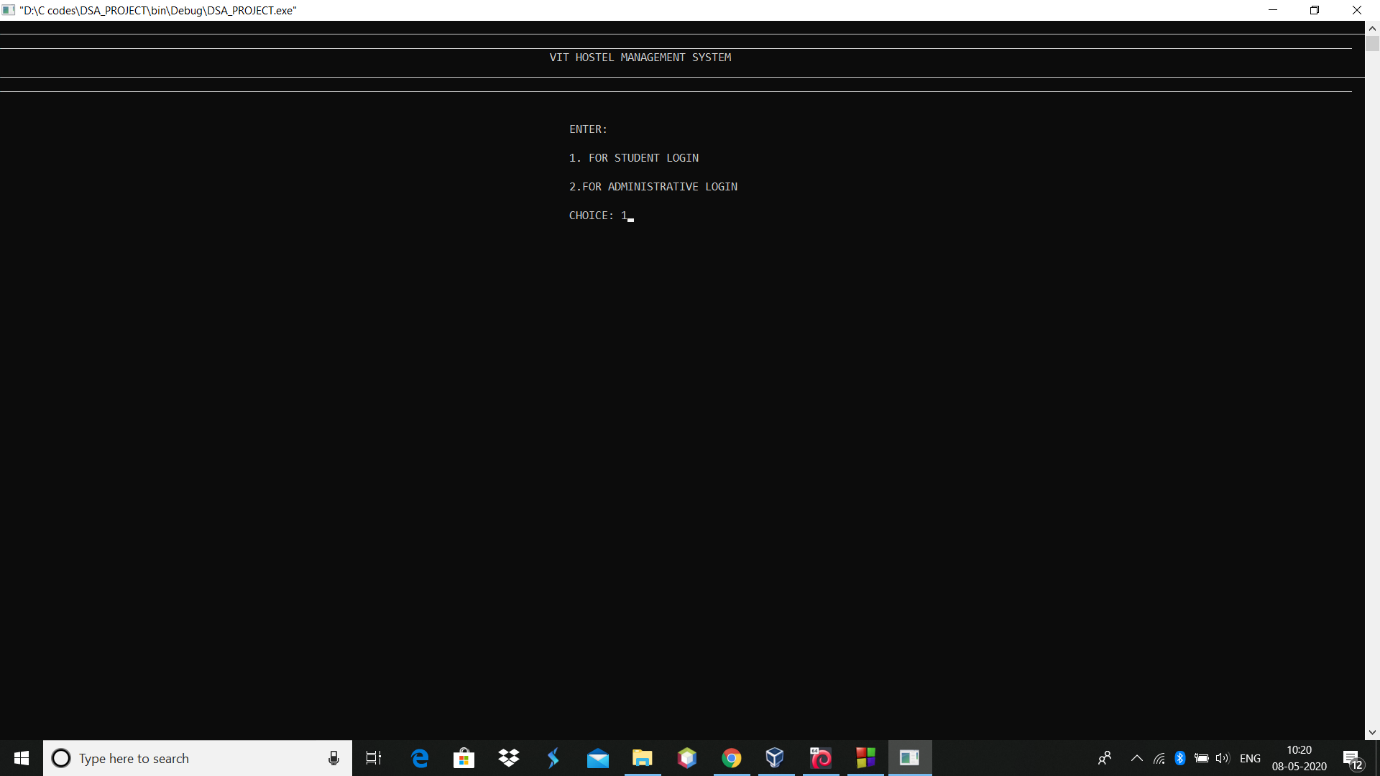
****

**TO ENTER THE STUDENT INFORMATION BY HOSTEL ADMINISTRATOR**

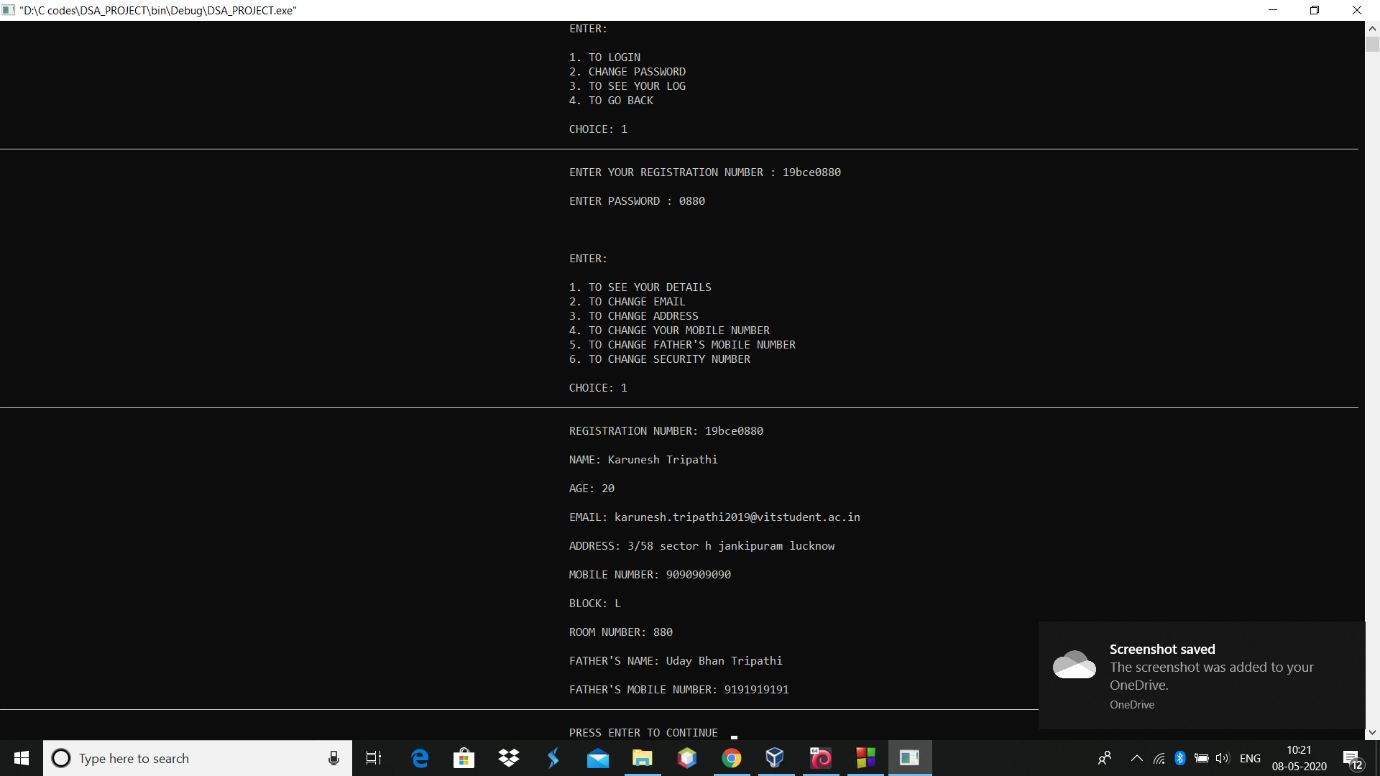
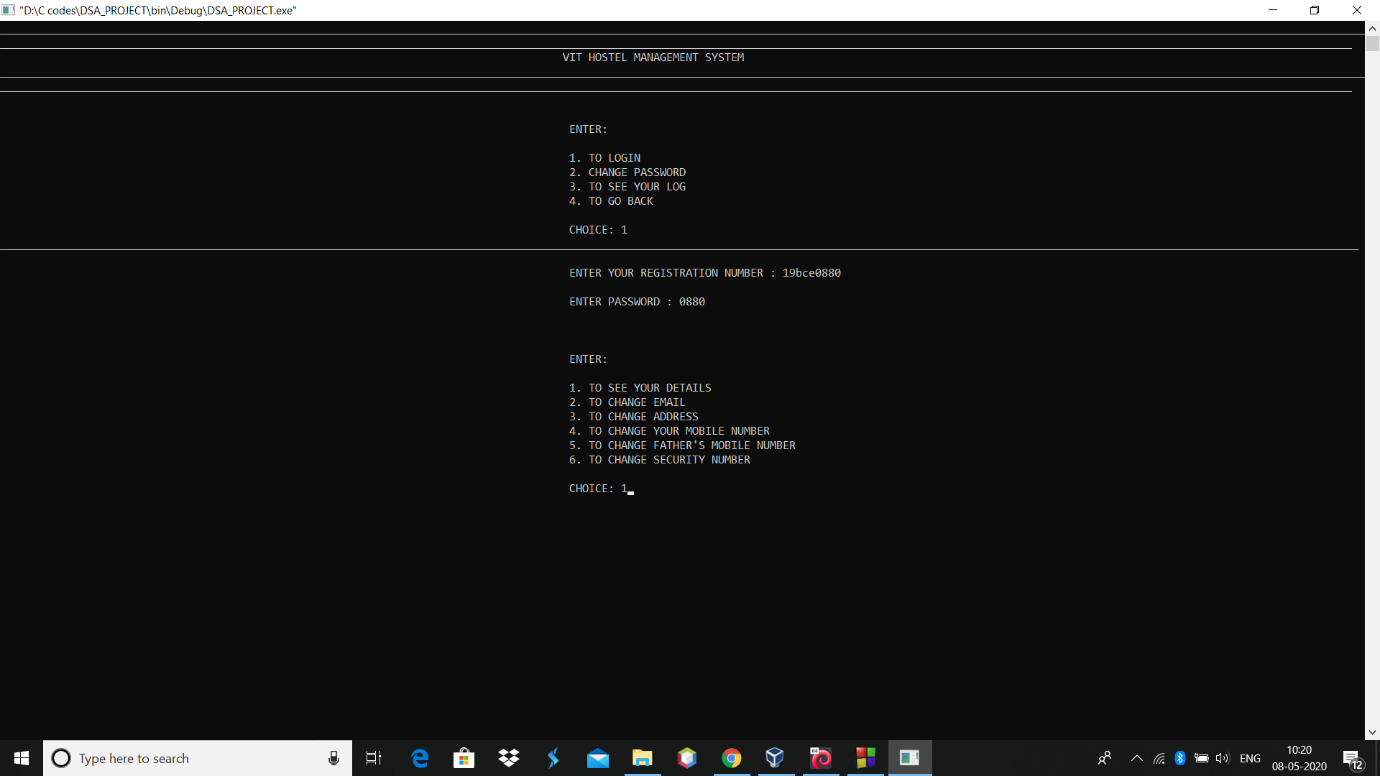
****

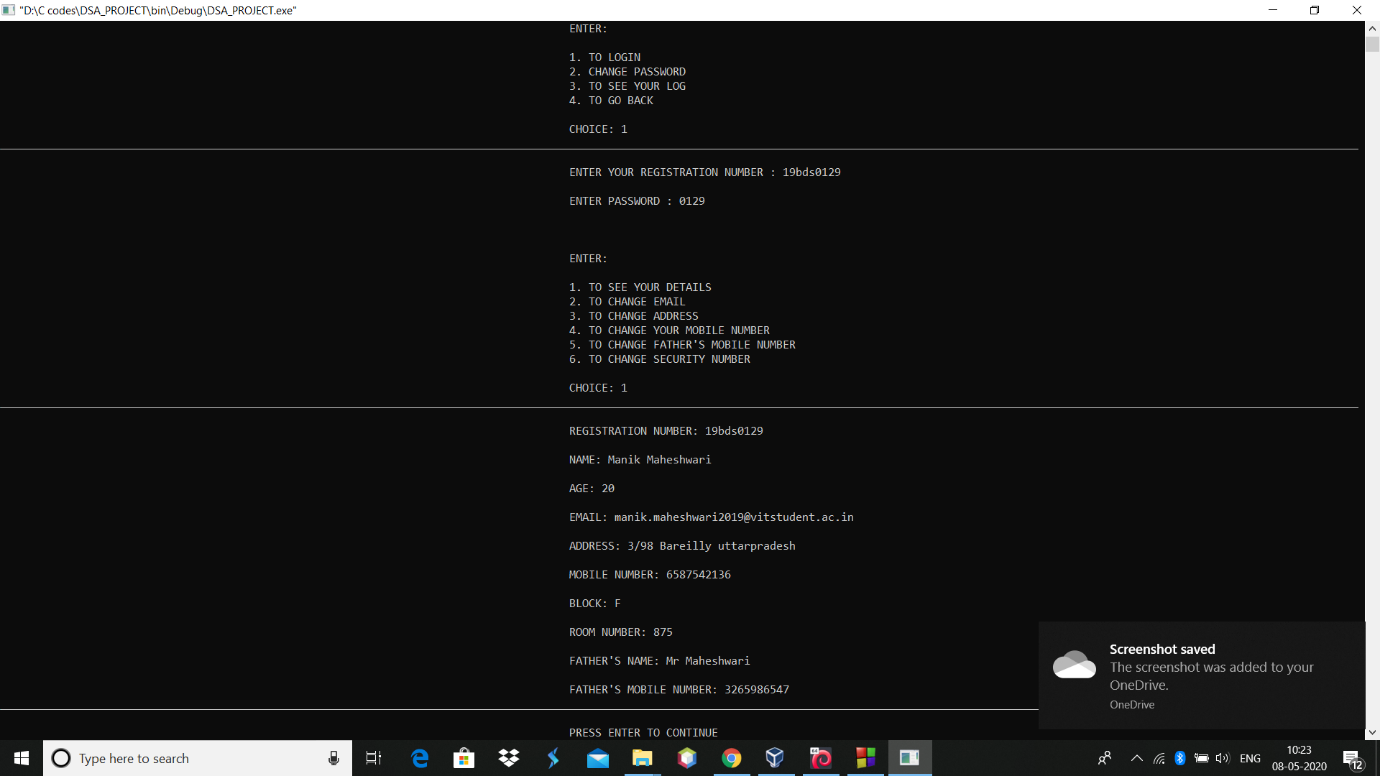
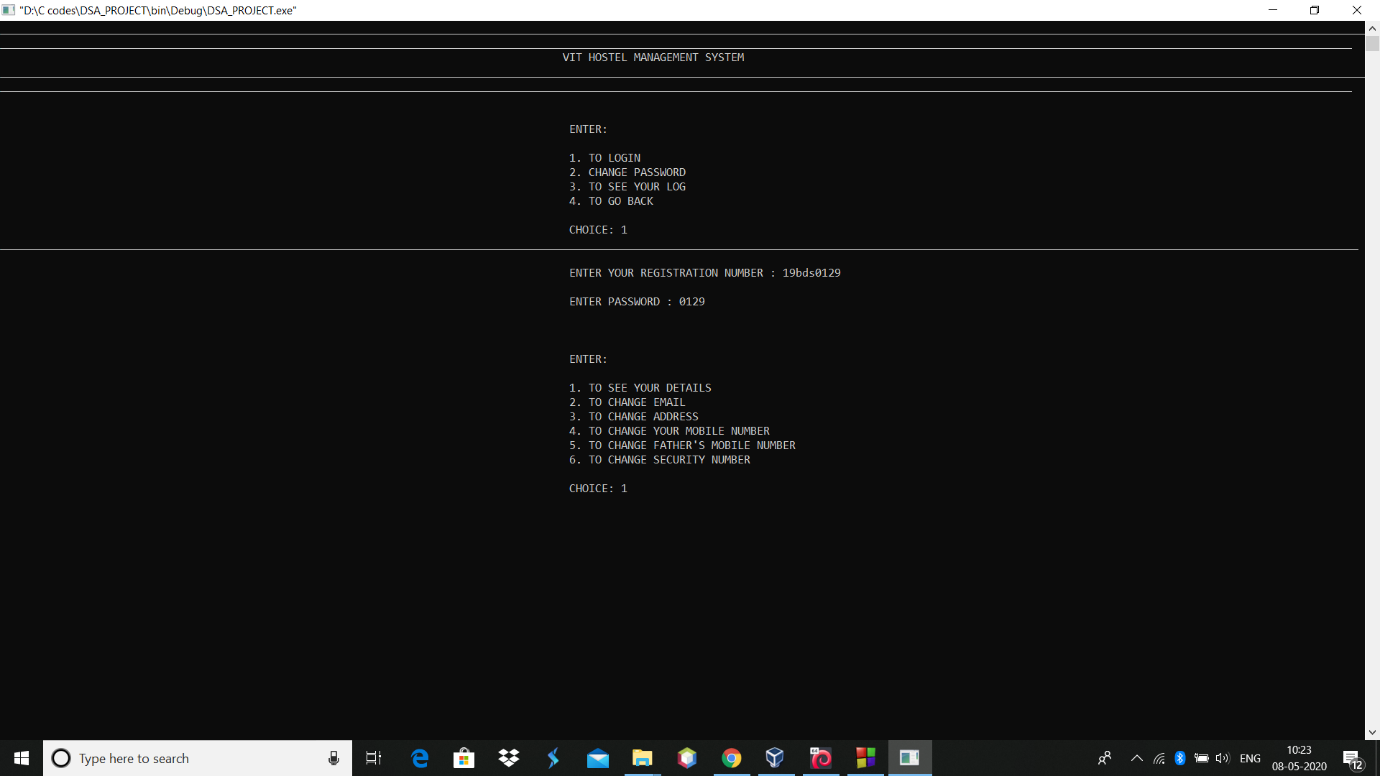
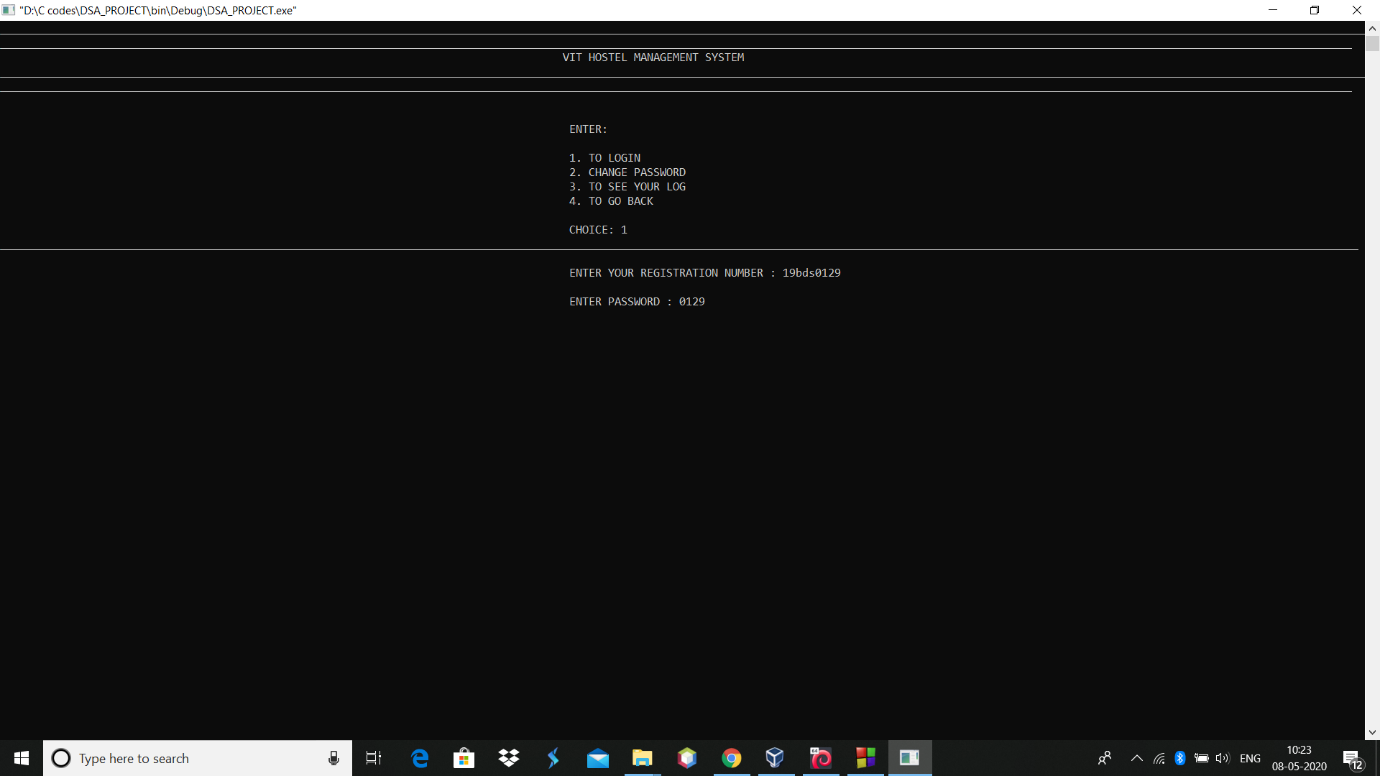
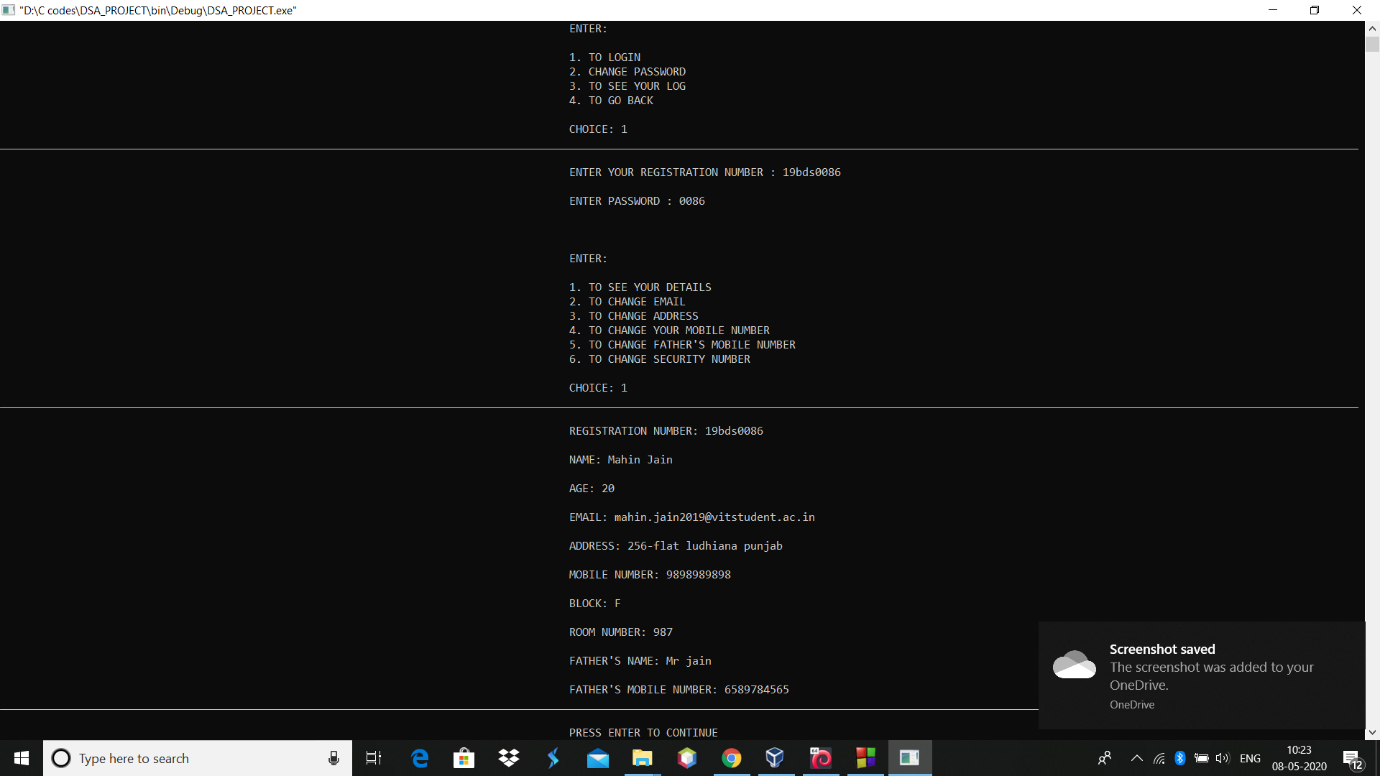
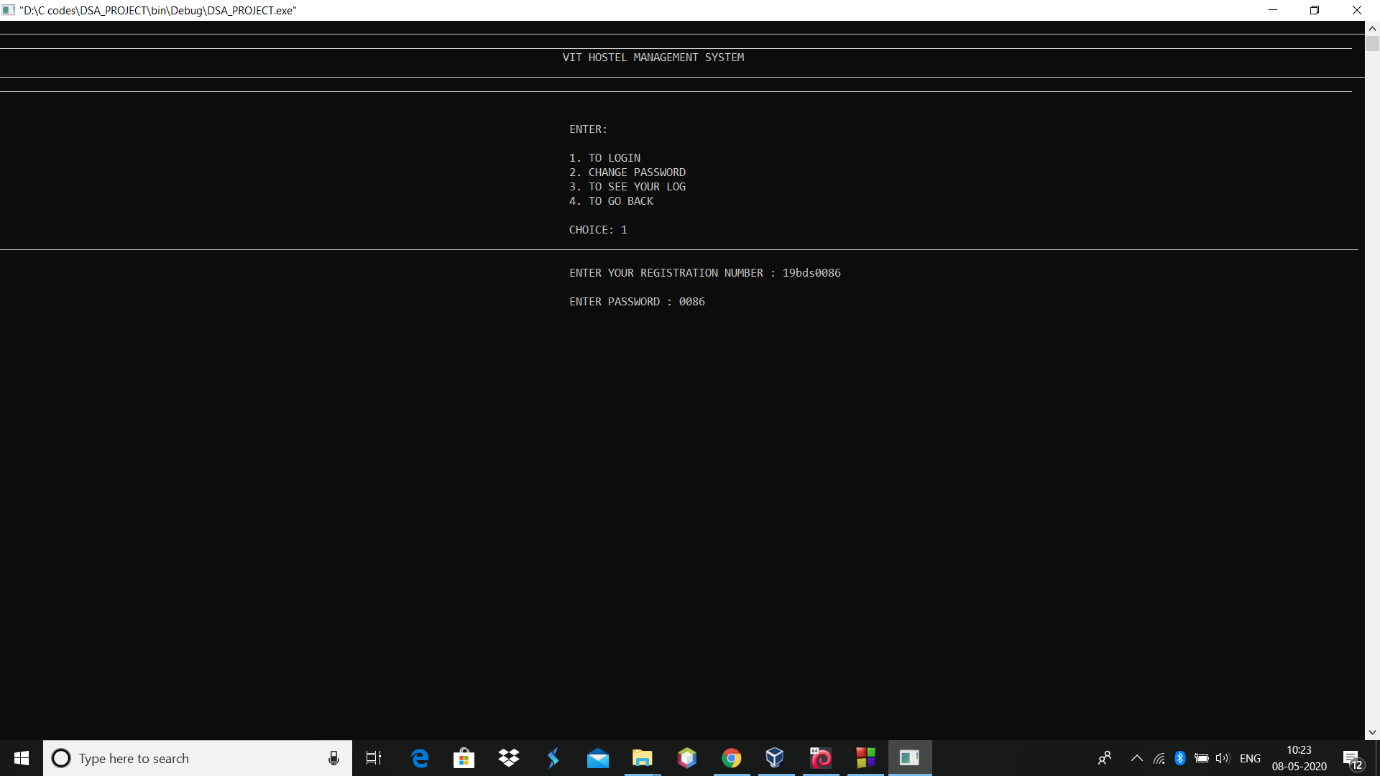
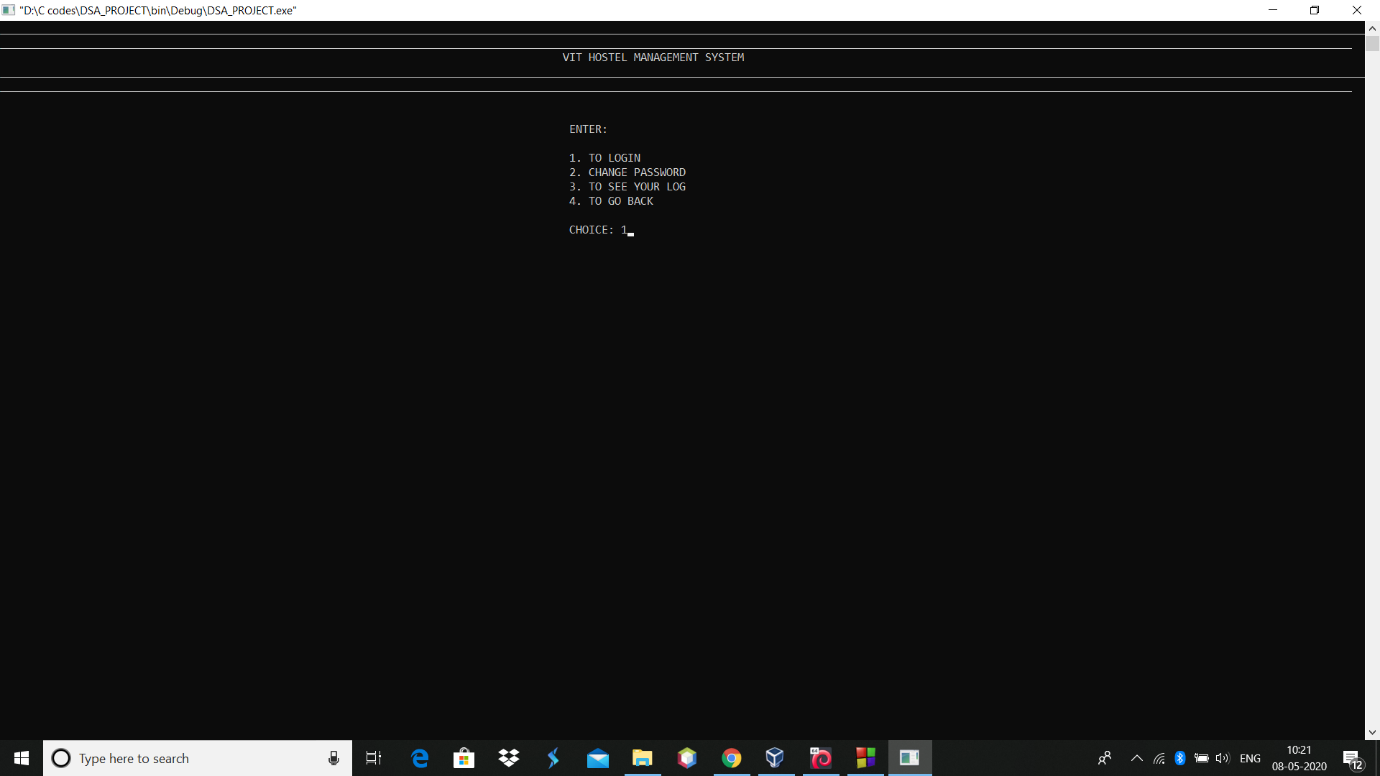
****

**TO LOGIN AS STUDENT**

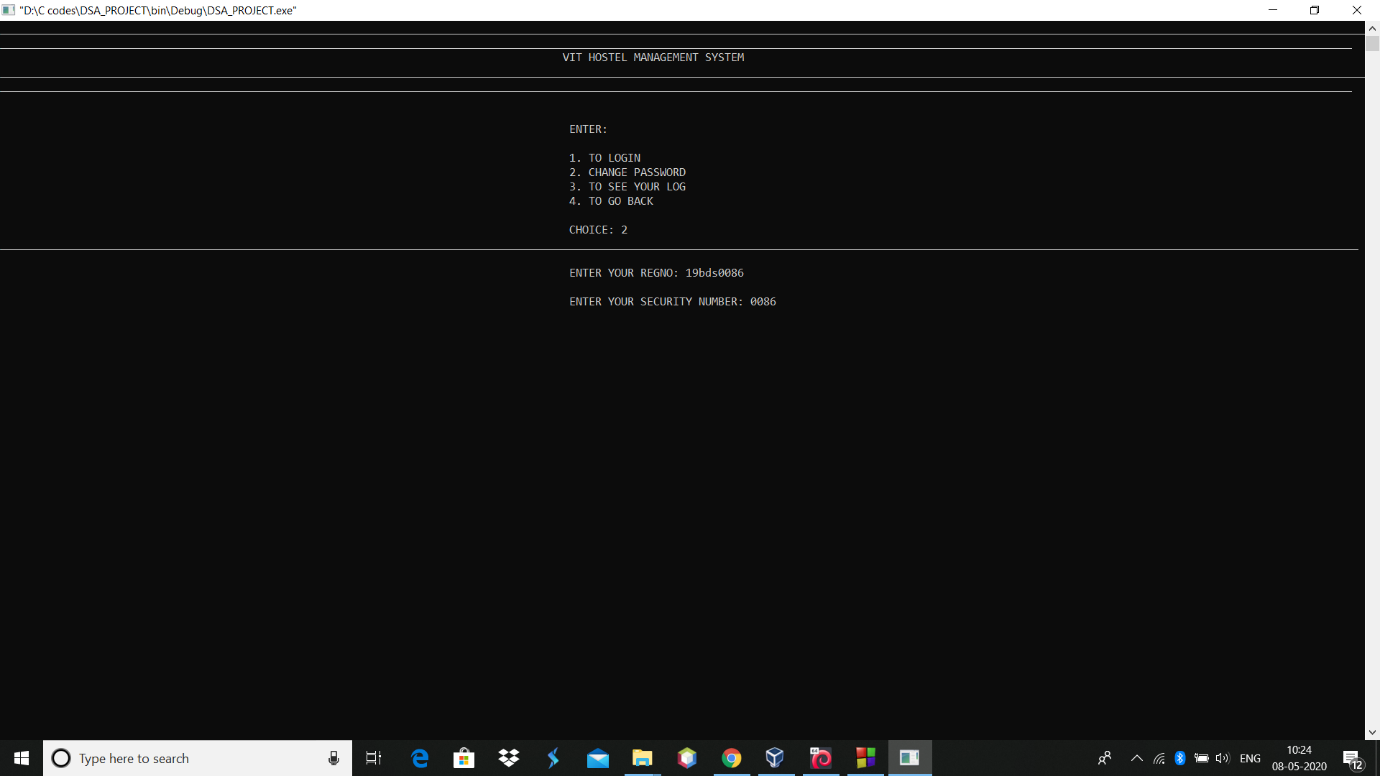
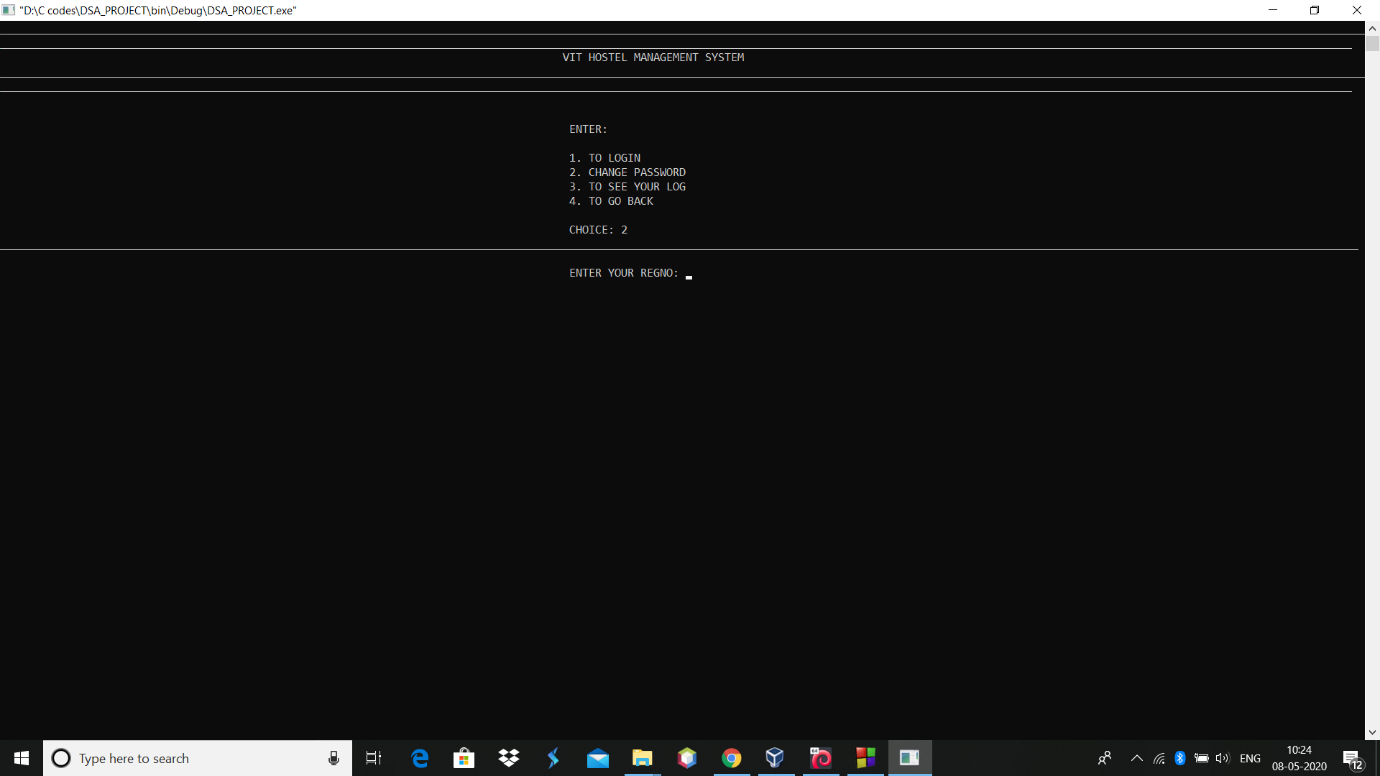
****

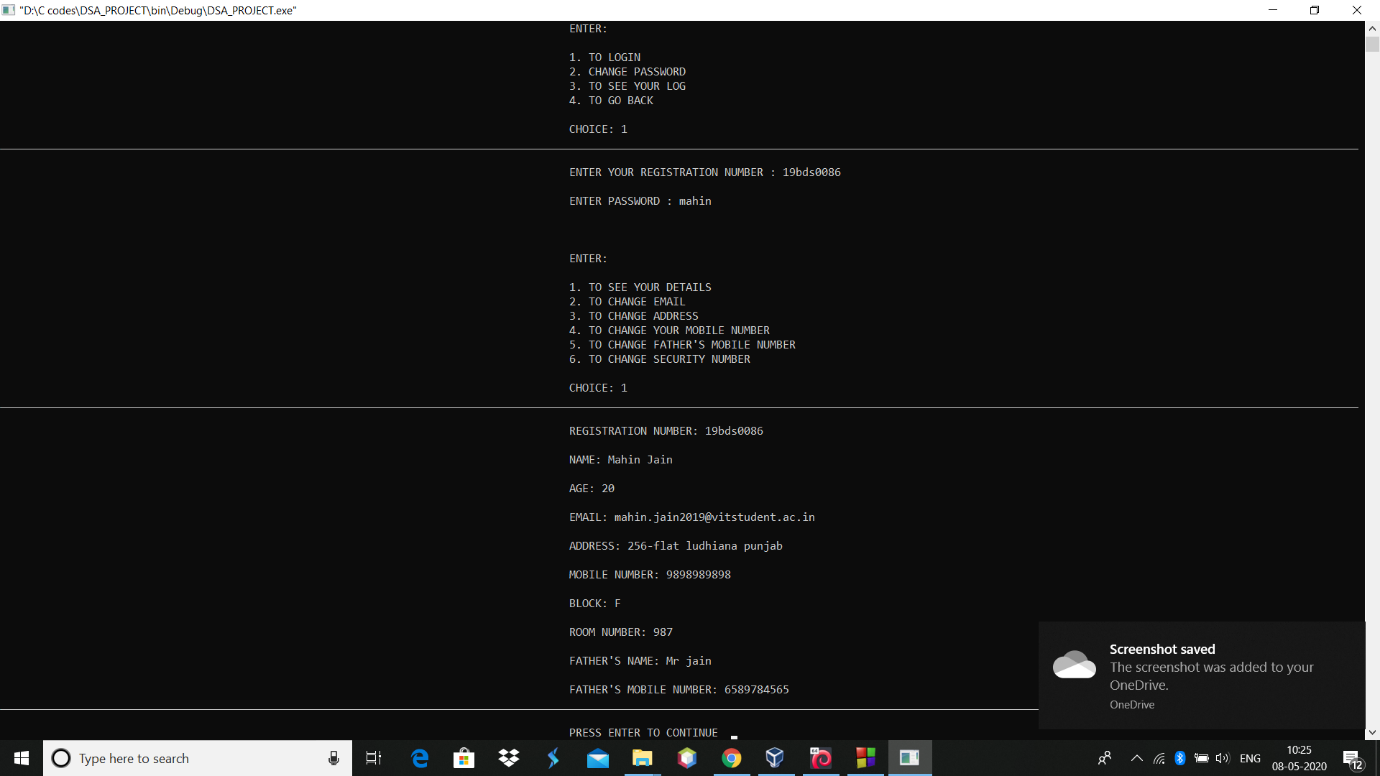
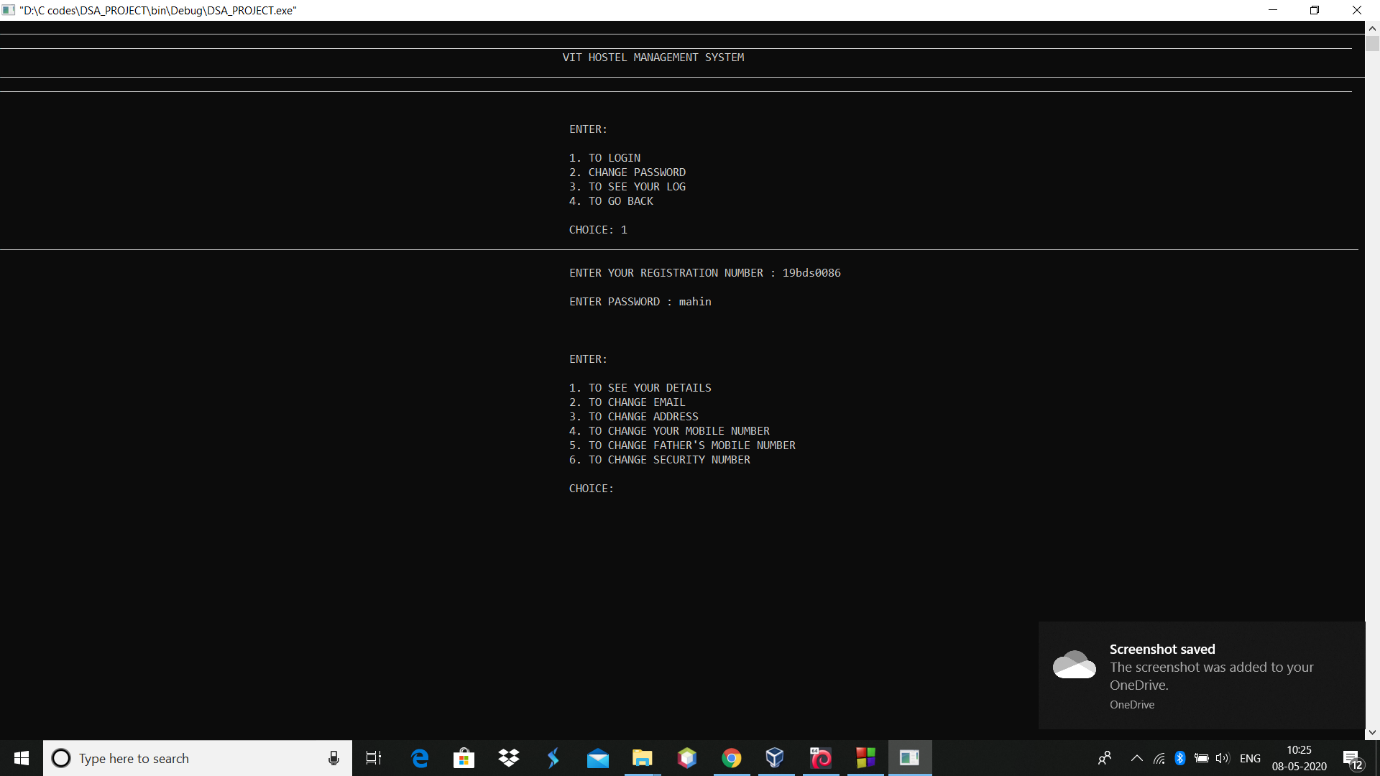
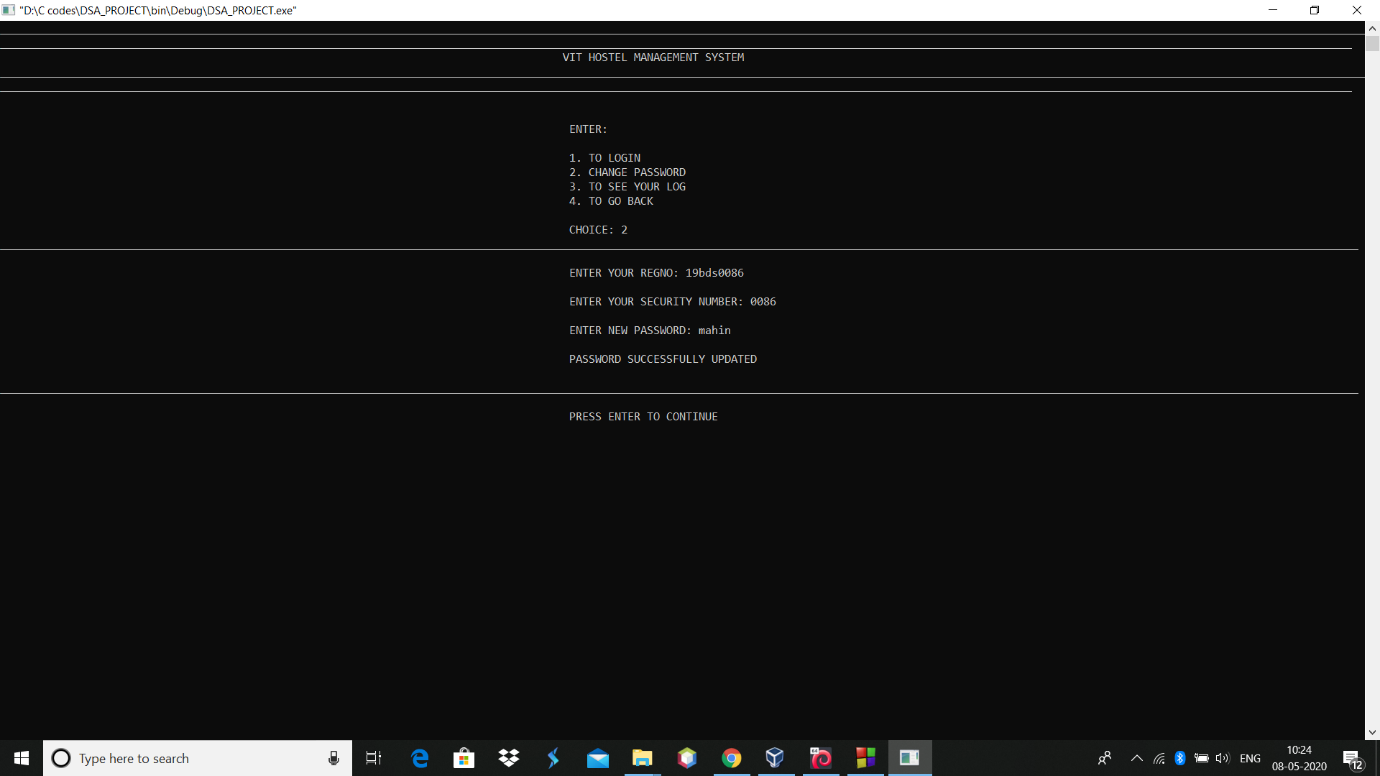
**TO SEE STUDENTS DETAILS**

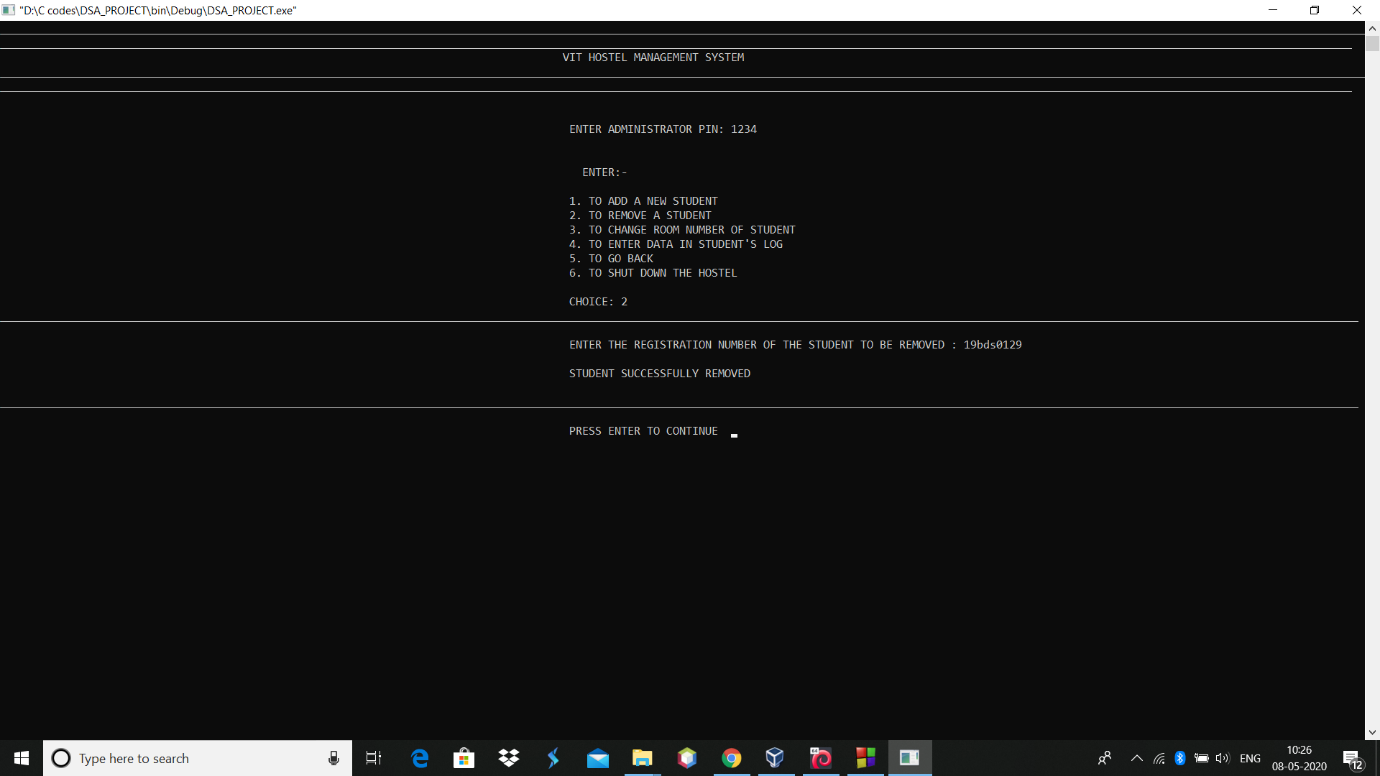
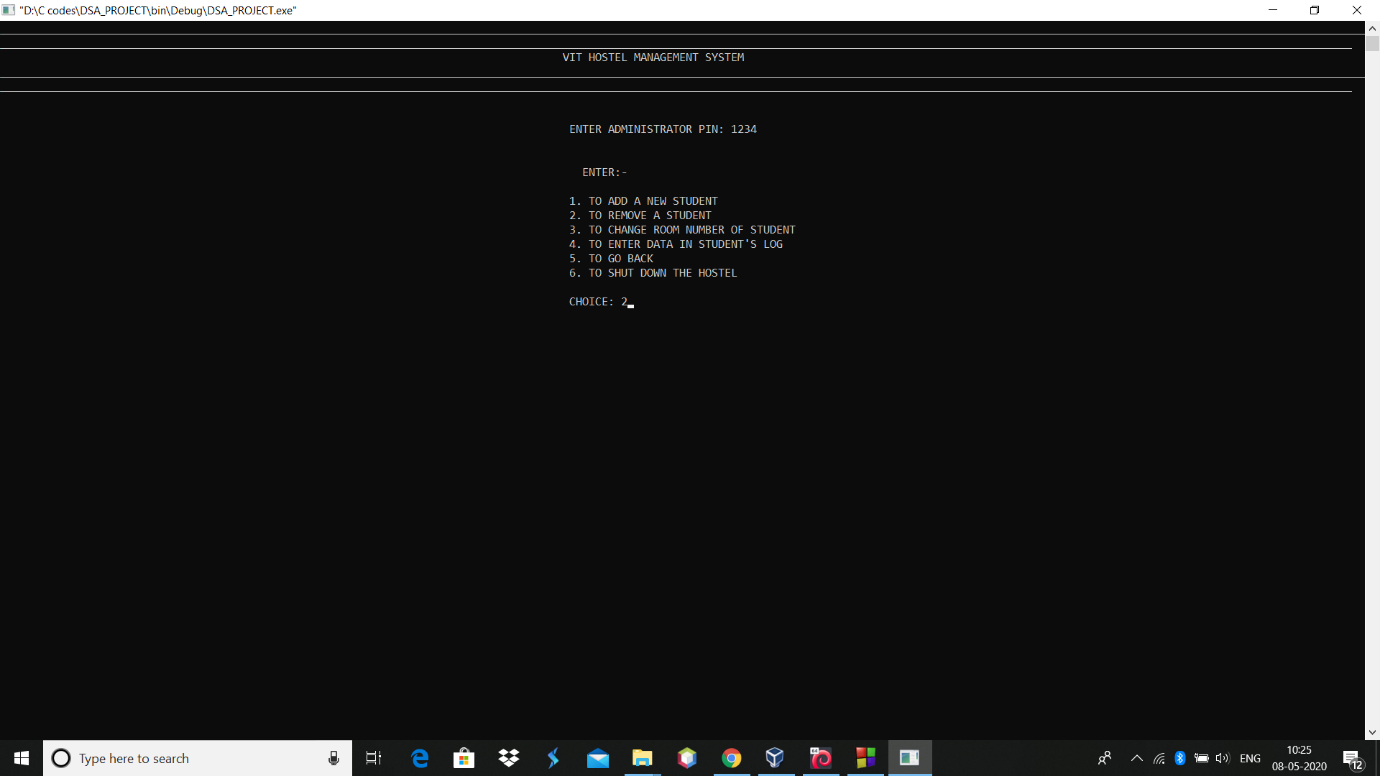
****

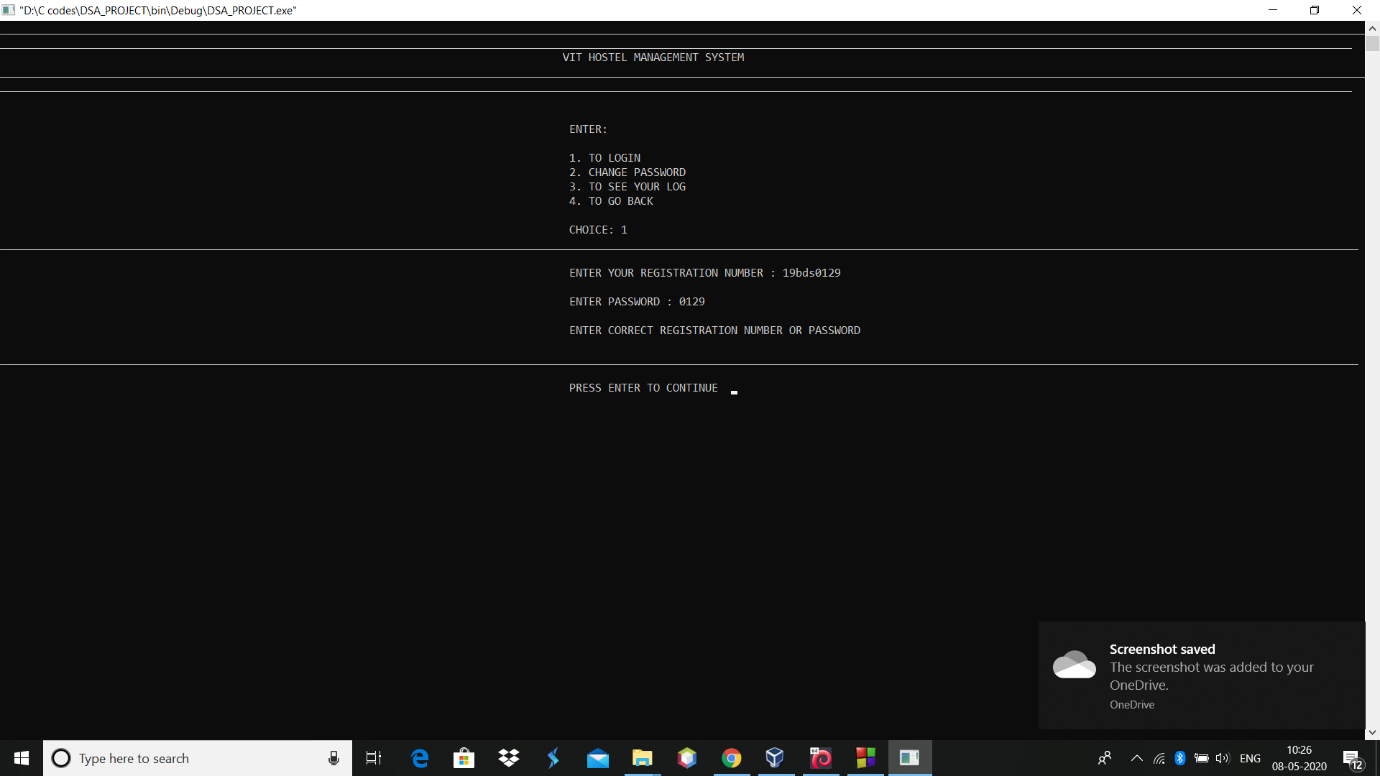
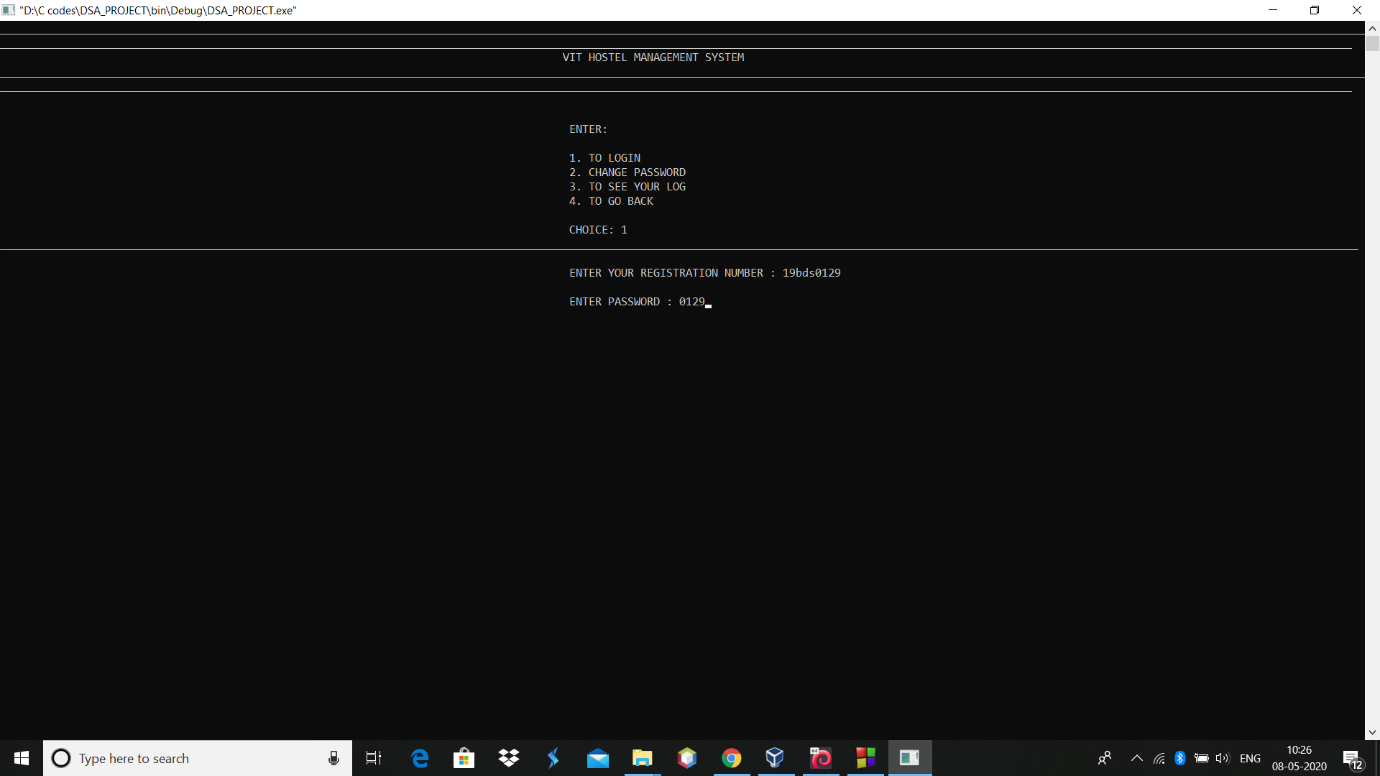
****

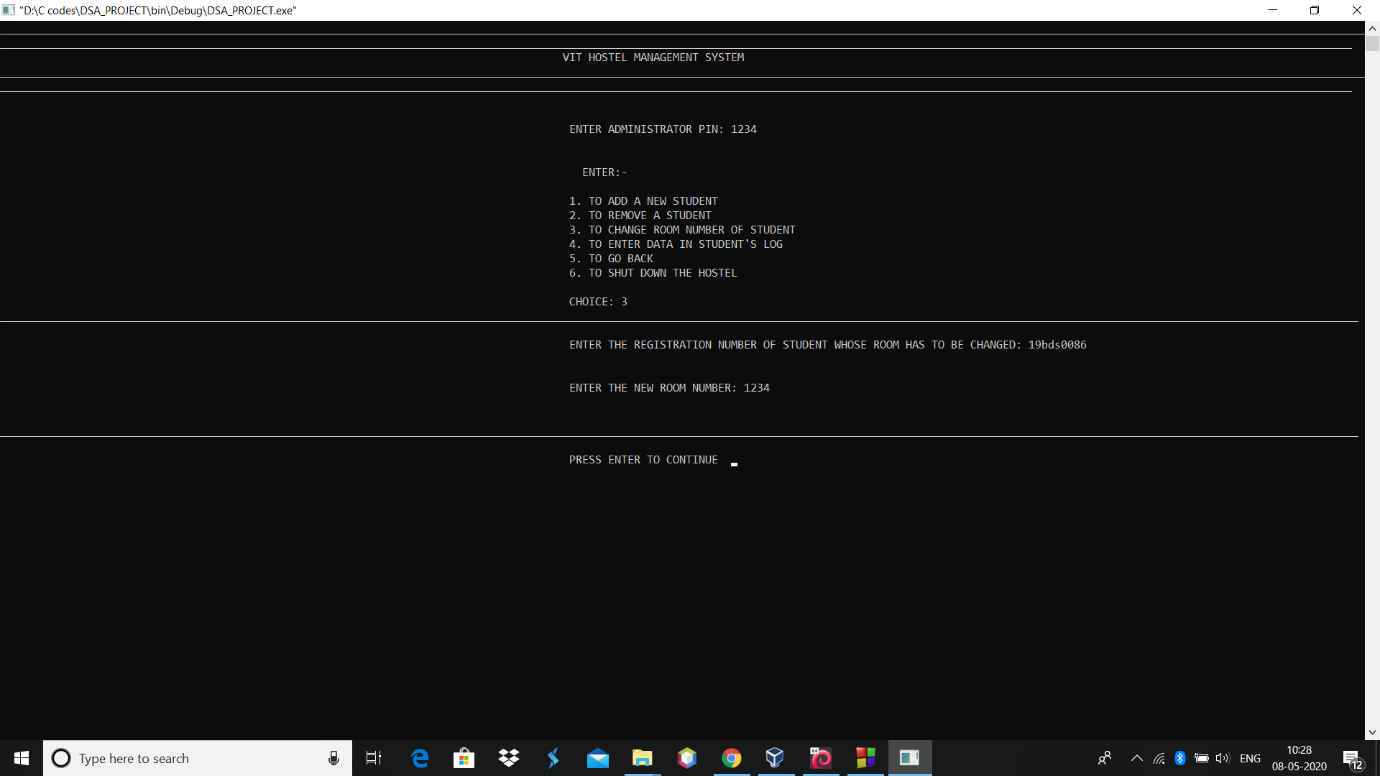
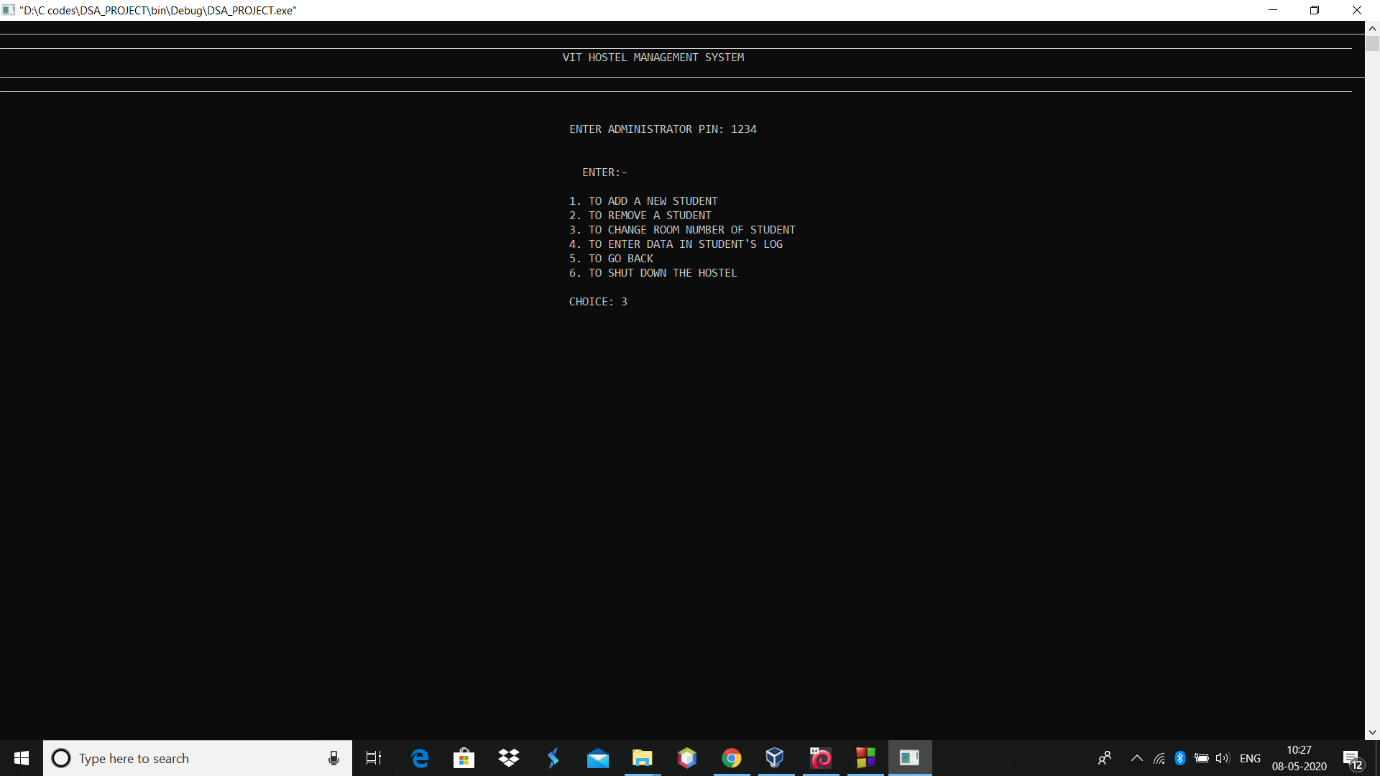
**TO CHANGE PASSWORD**

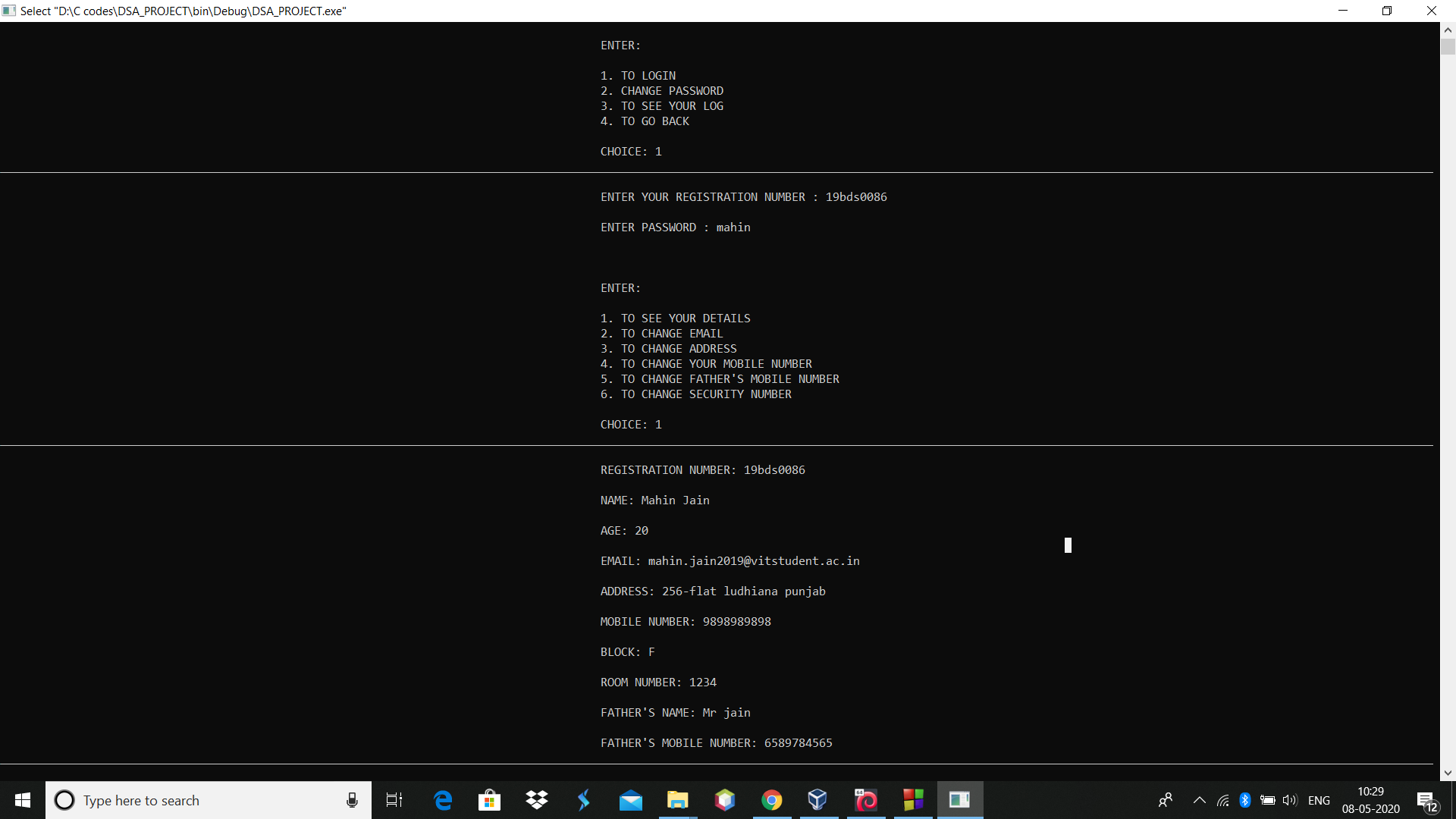
****

****

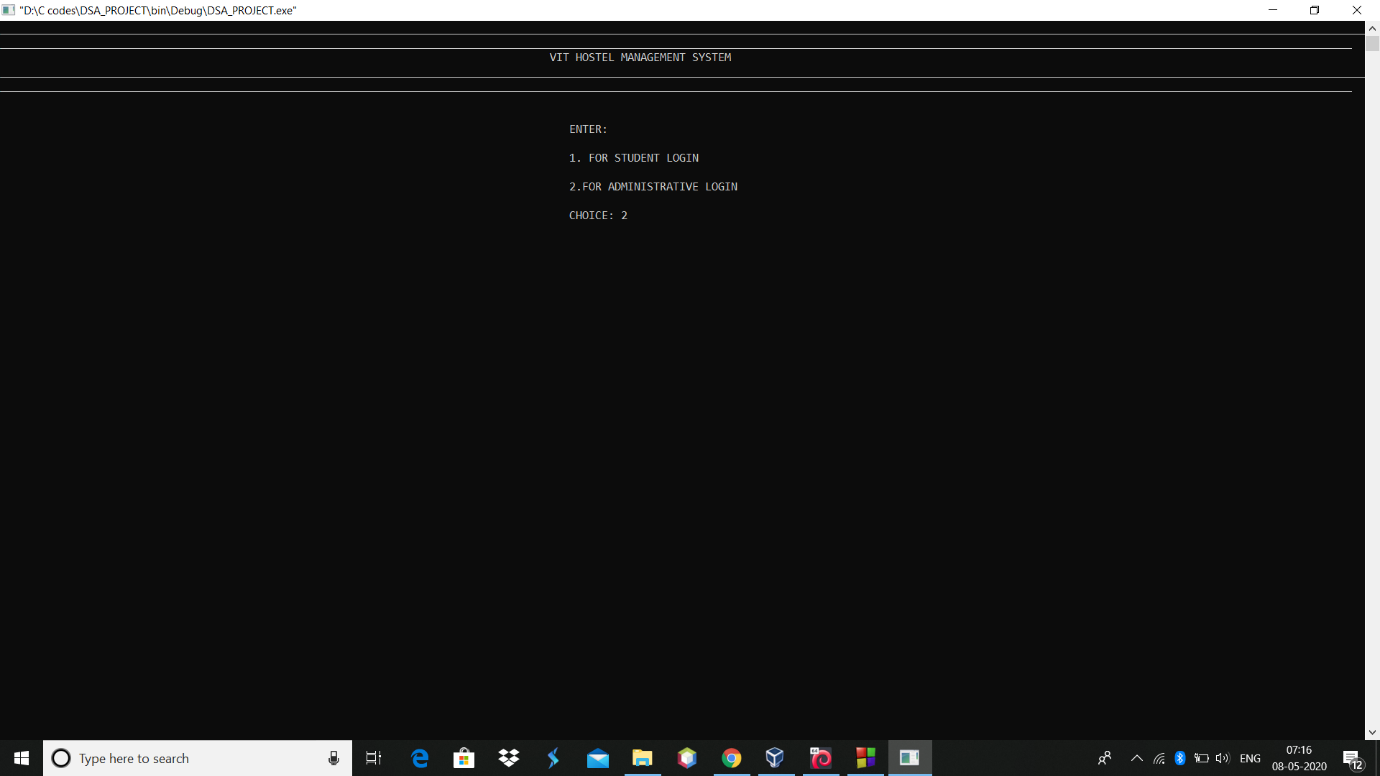
**TO REMOVE STUDENT**

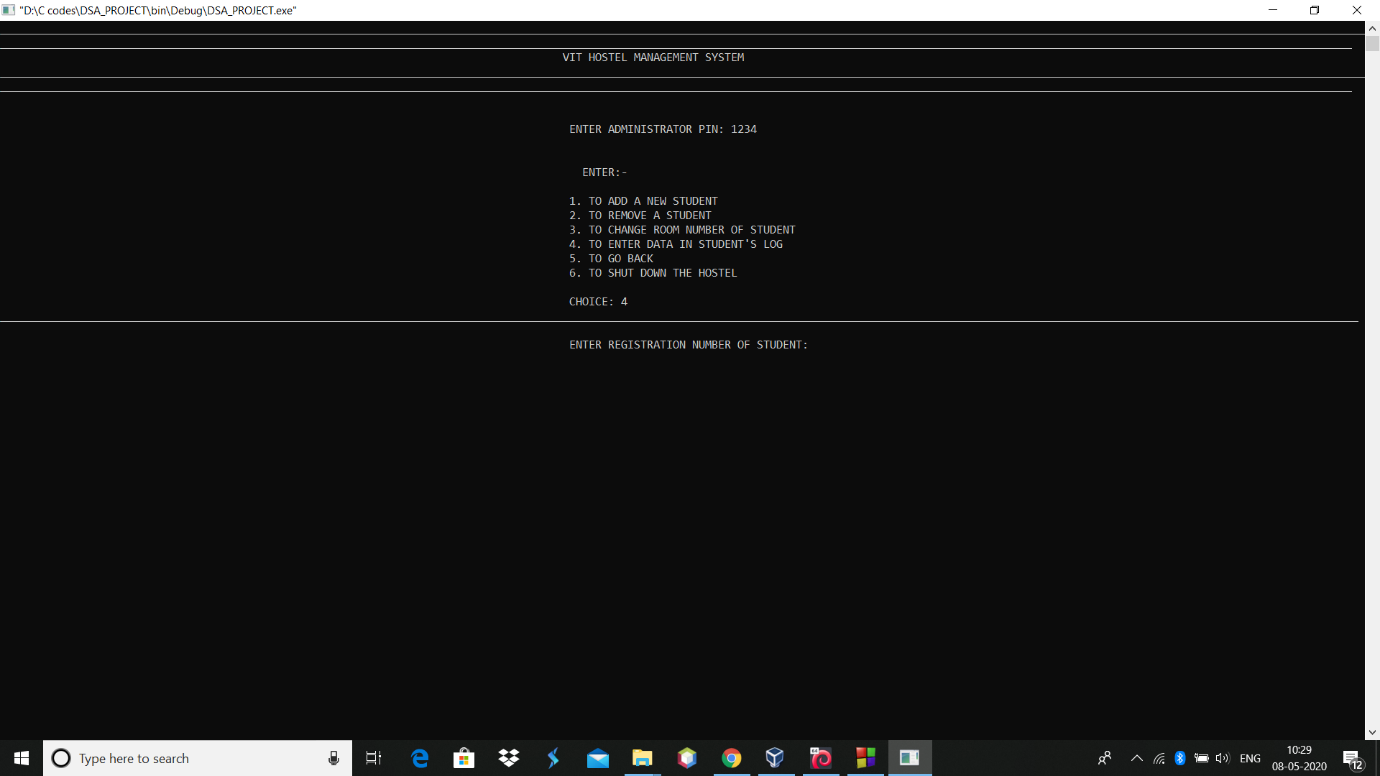
****

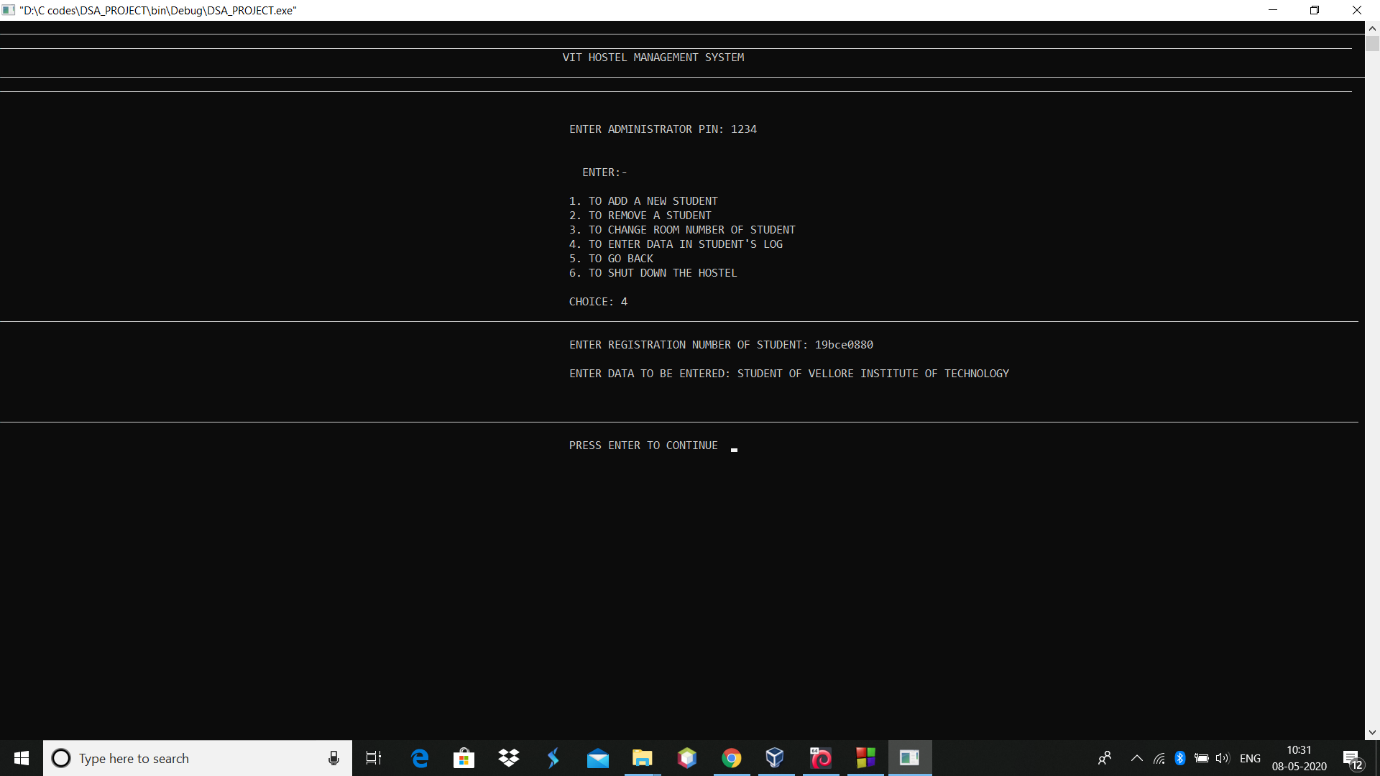
**TO CHANGE ROOM NUMBER**

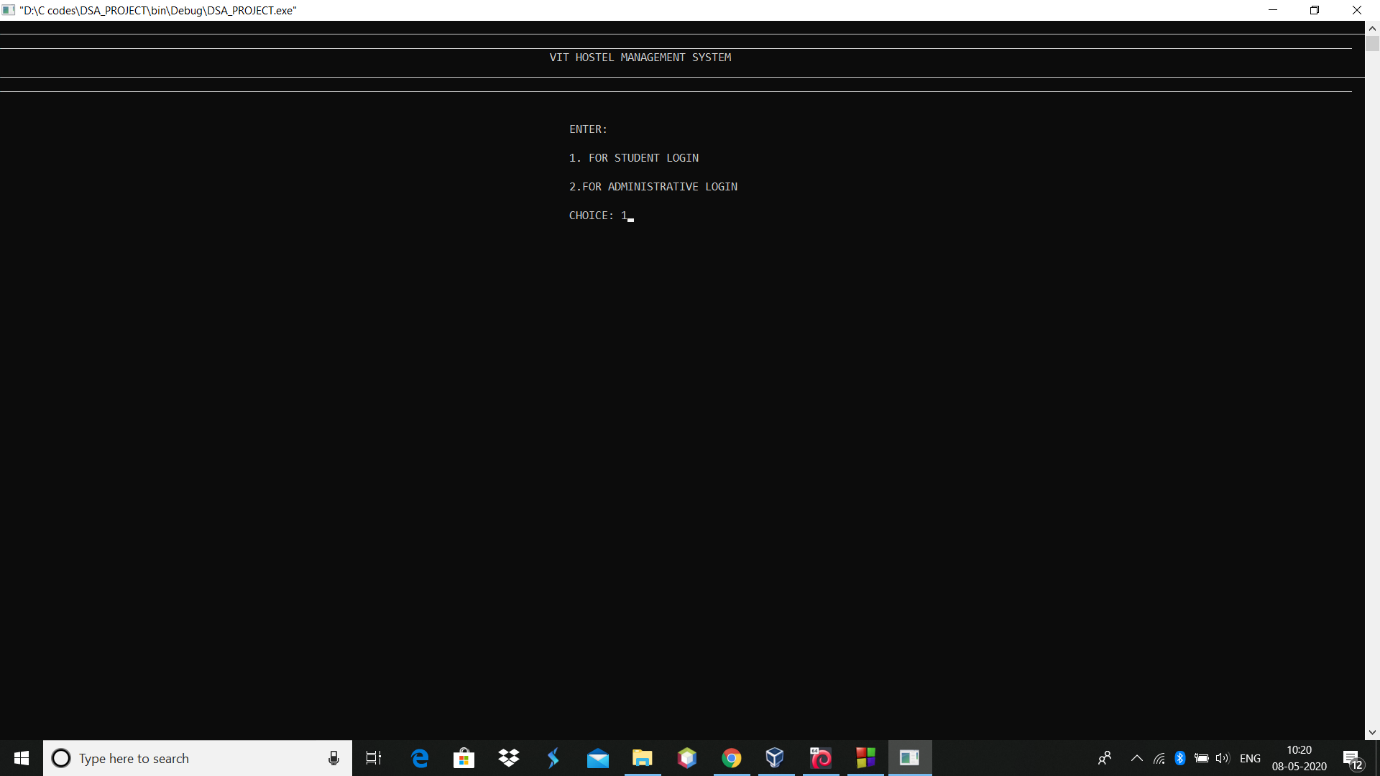
****

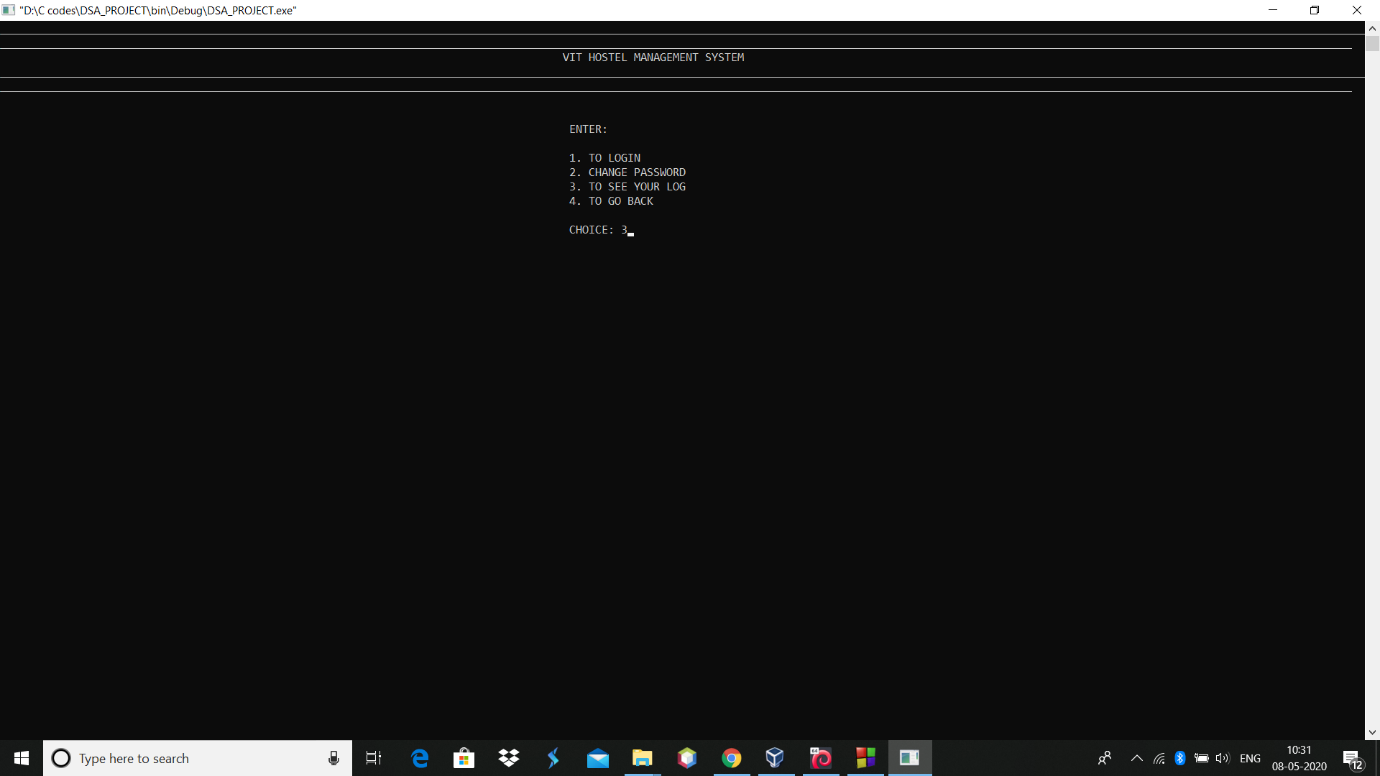
**TO ENTER DATA IN STUDENT’S LOG**

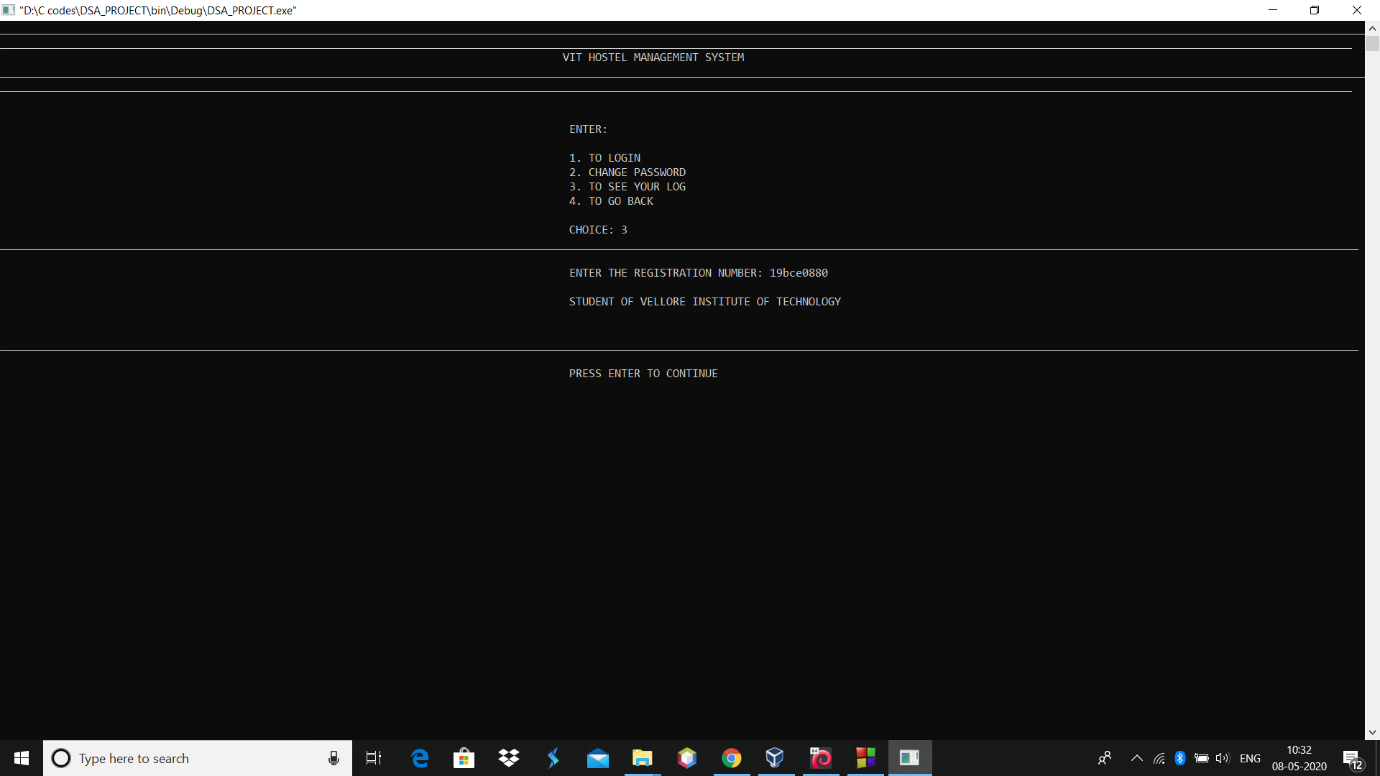
****

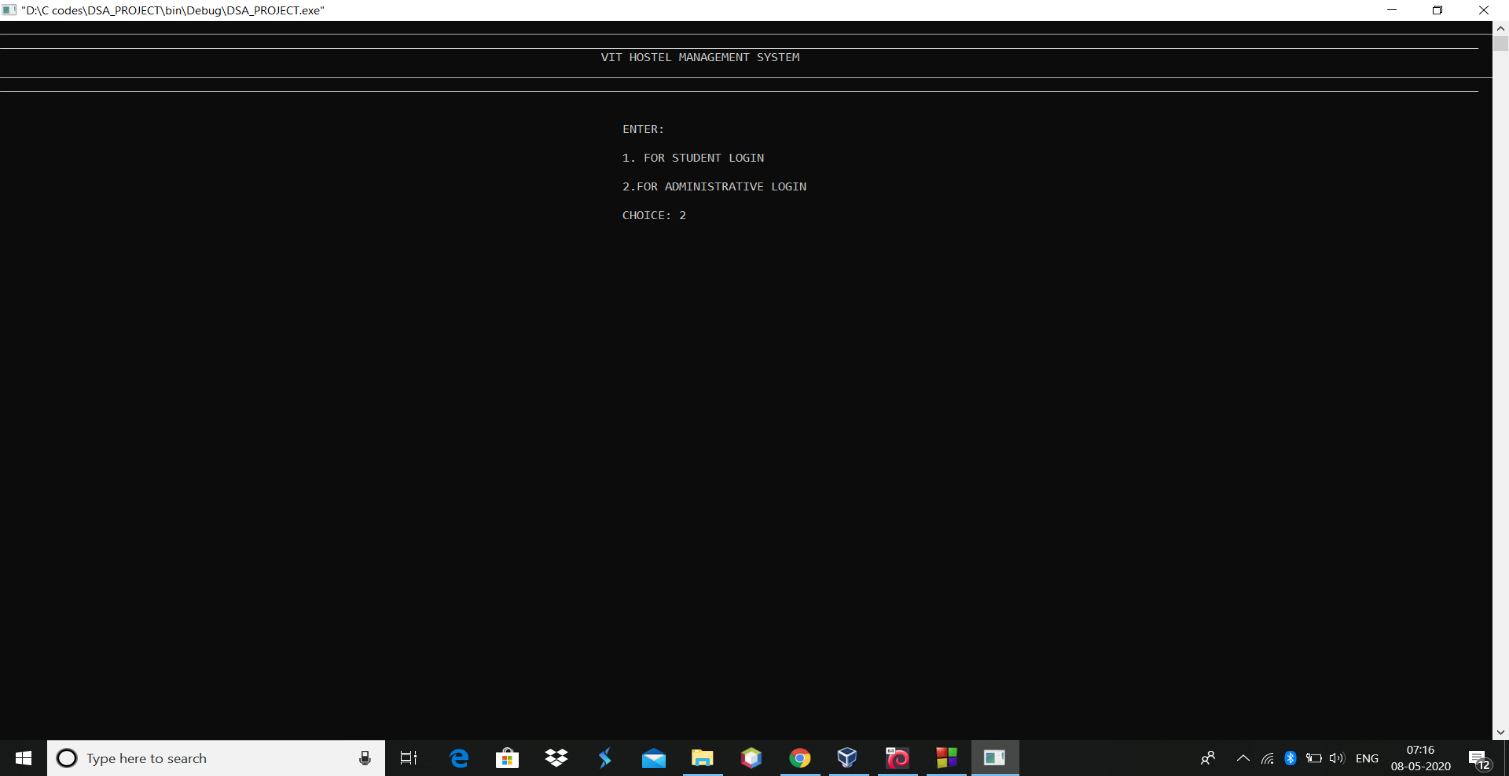
****

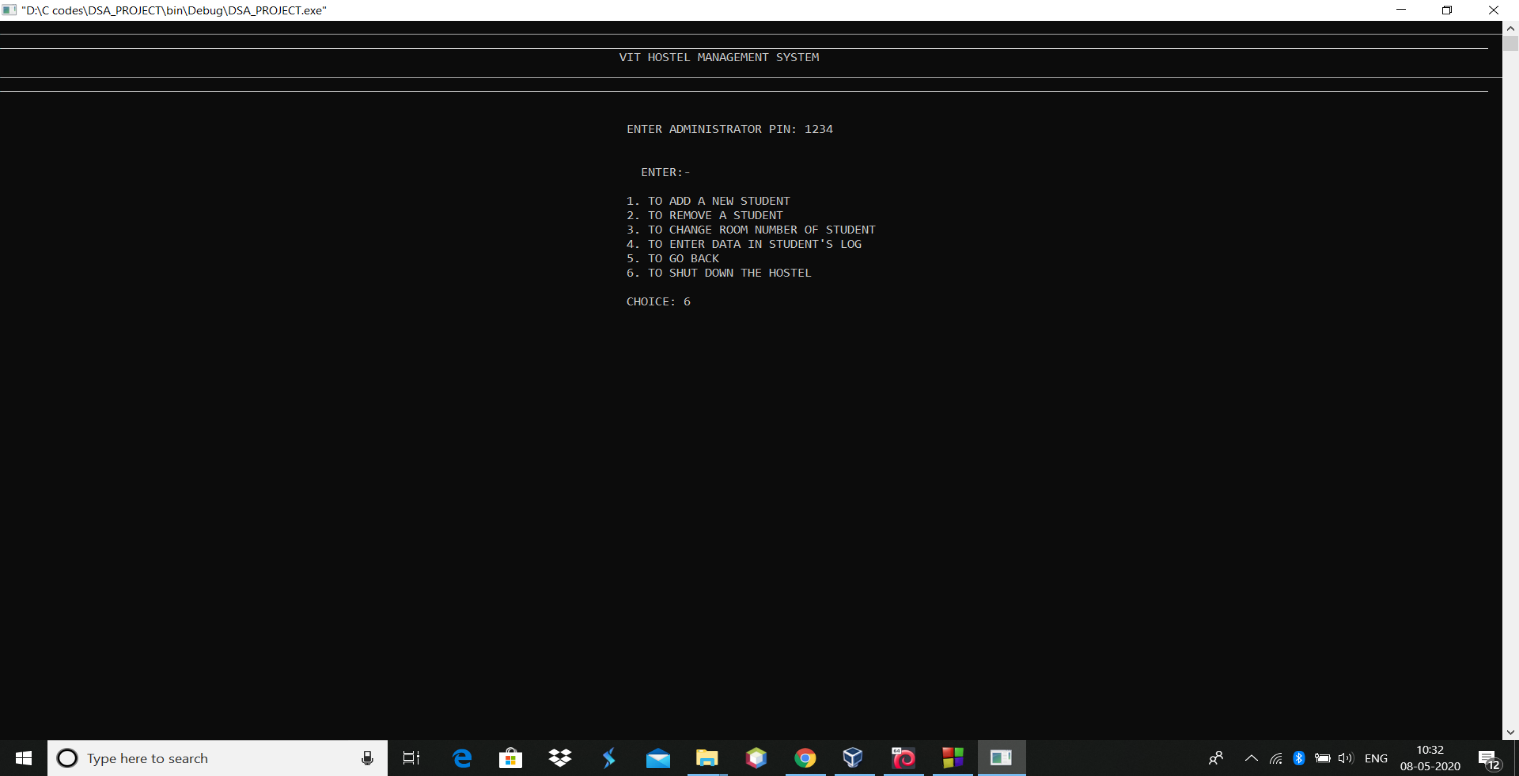
****

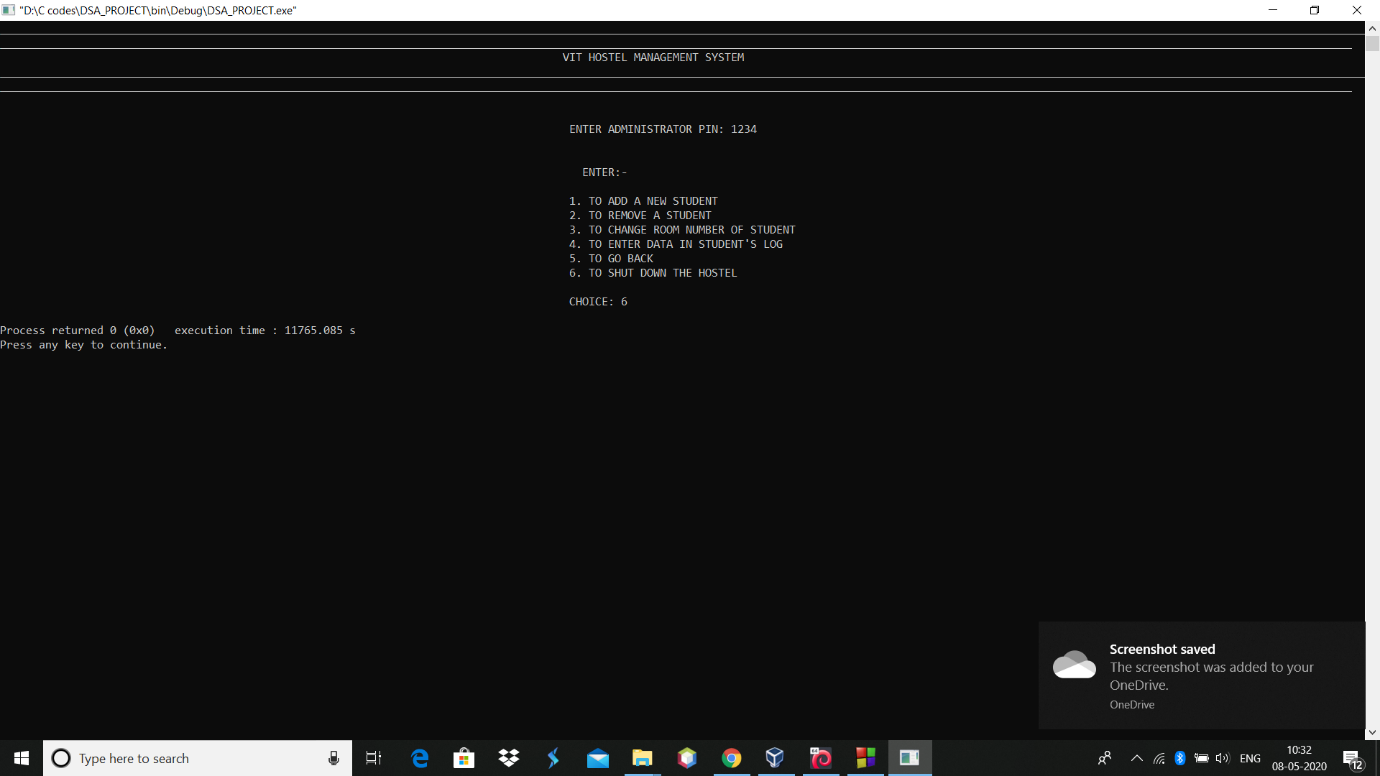
**TO SEE STUDENT’S LOG**

****

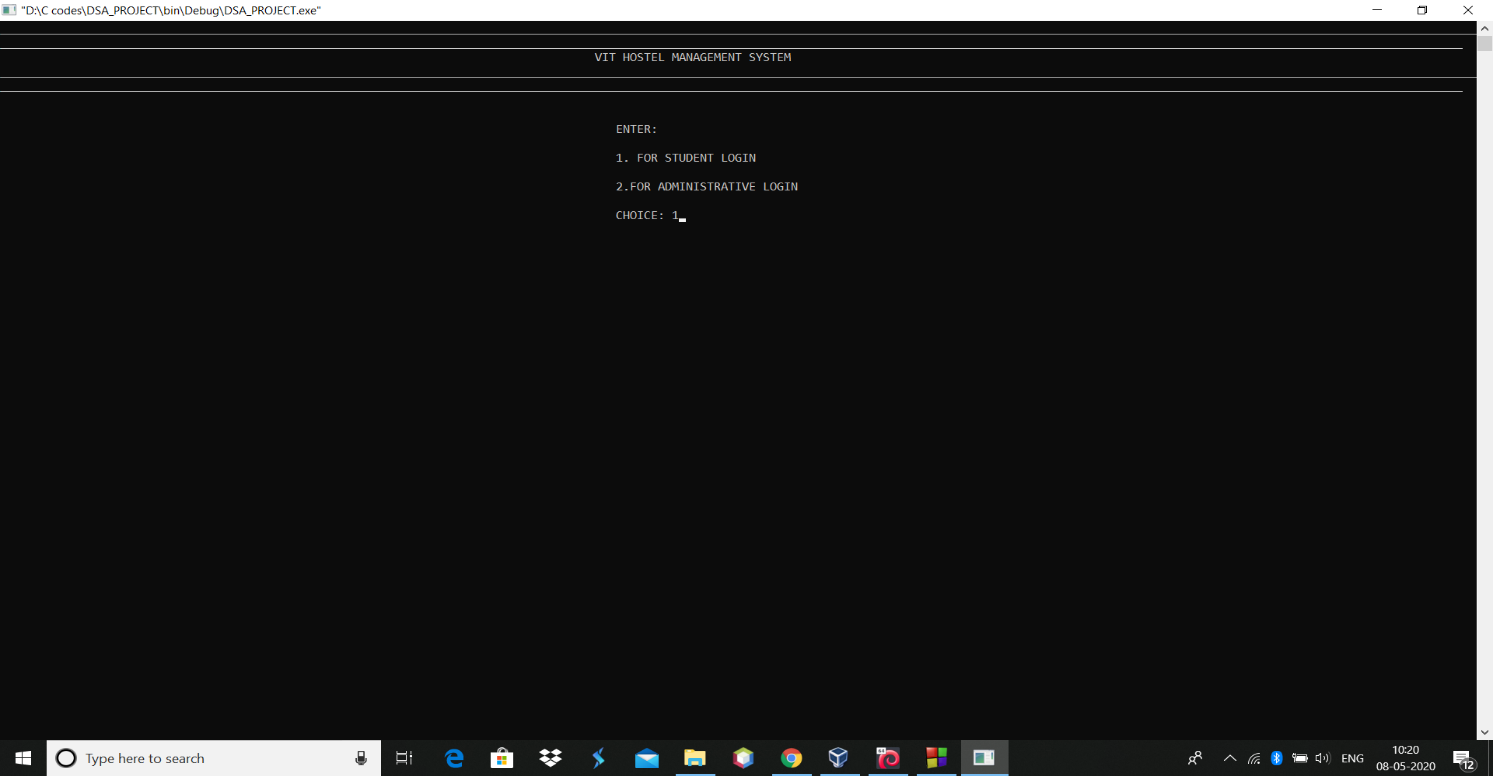
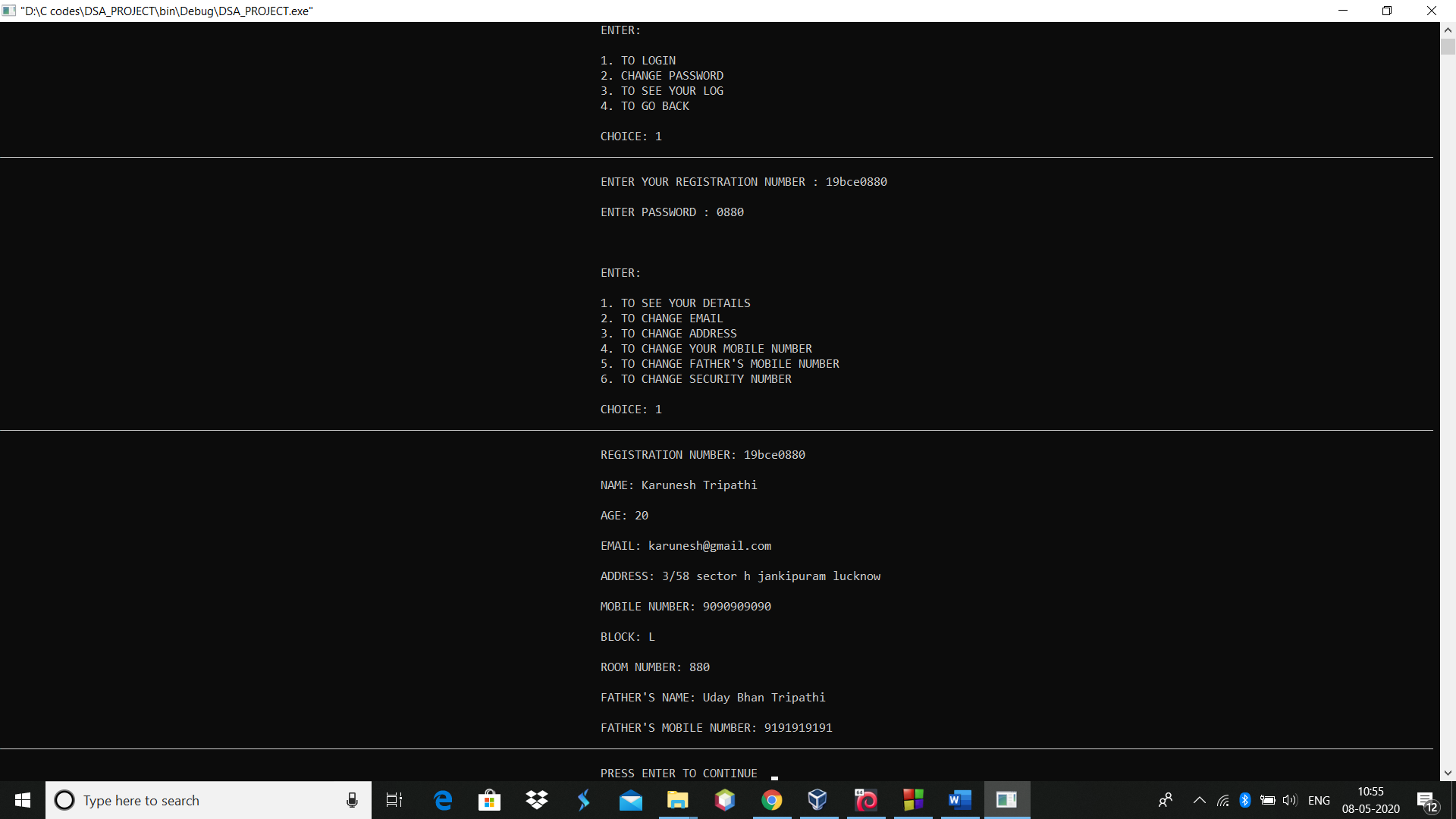
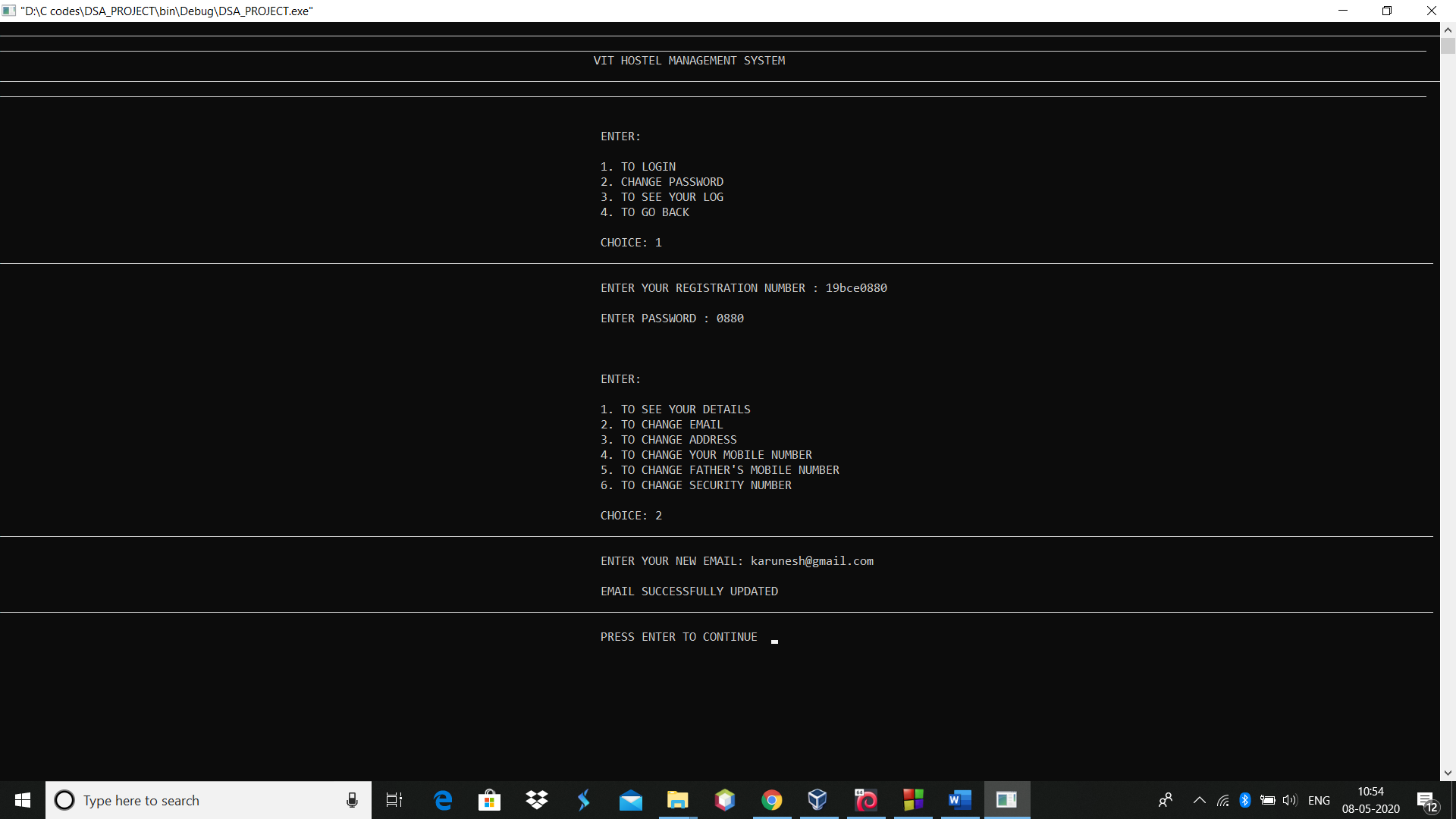
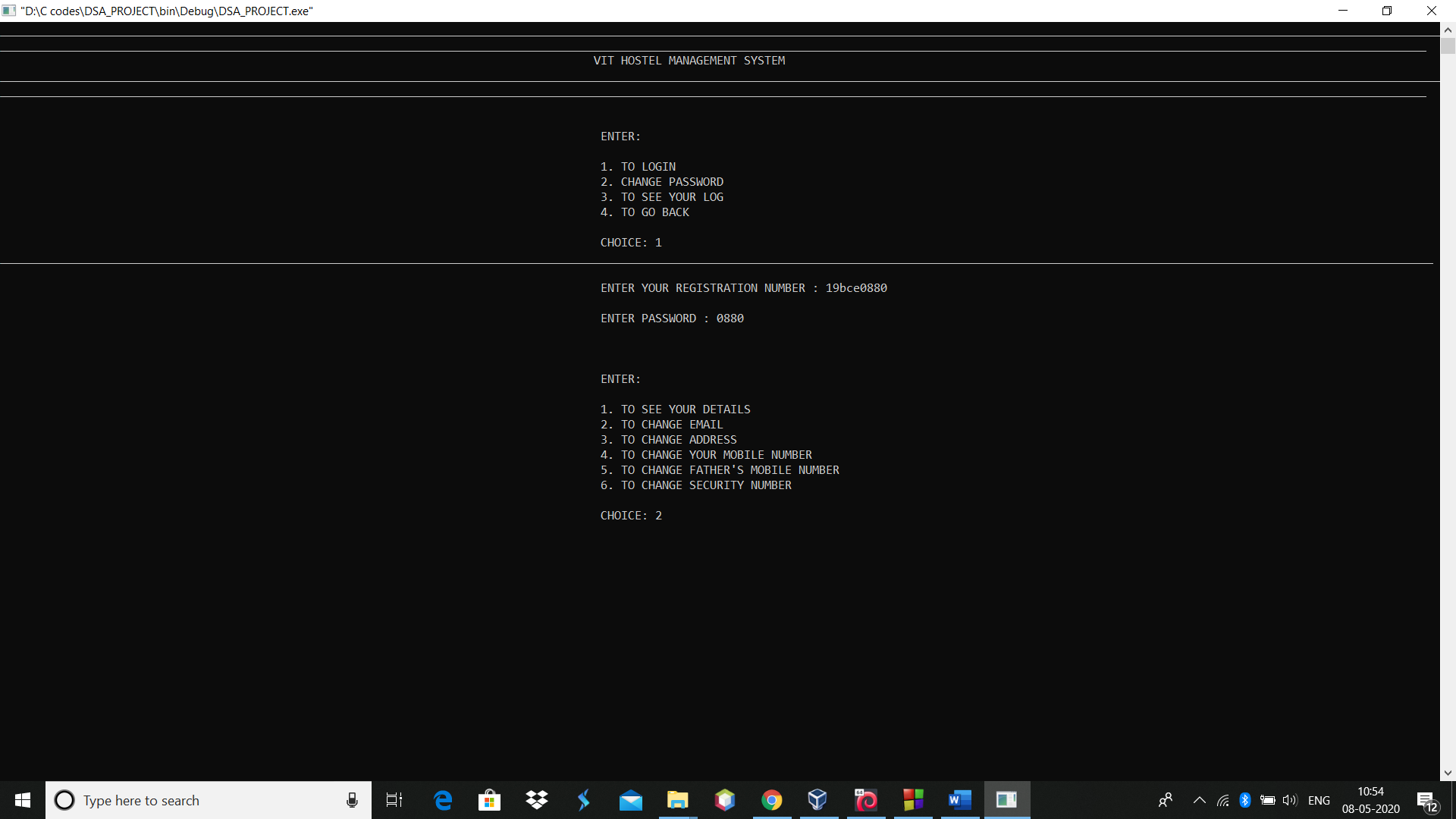
****

**TO SHUT DOWN HOSTEL**

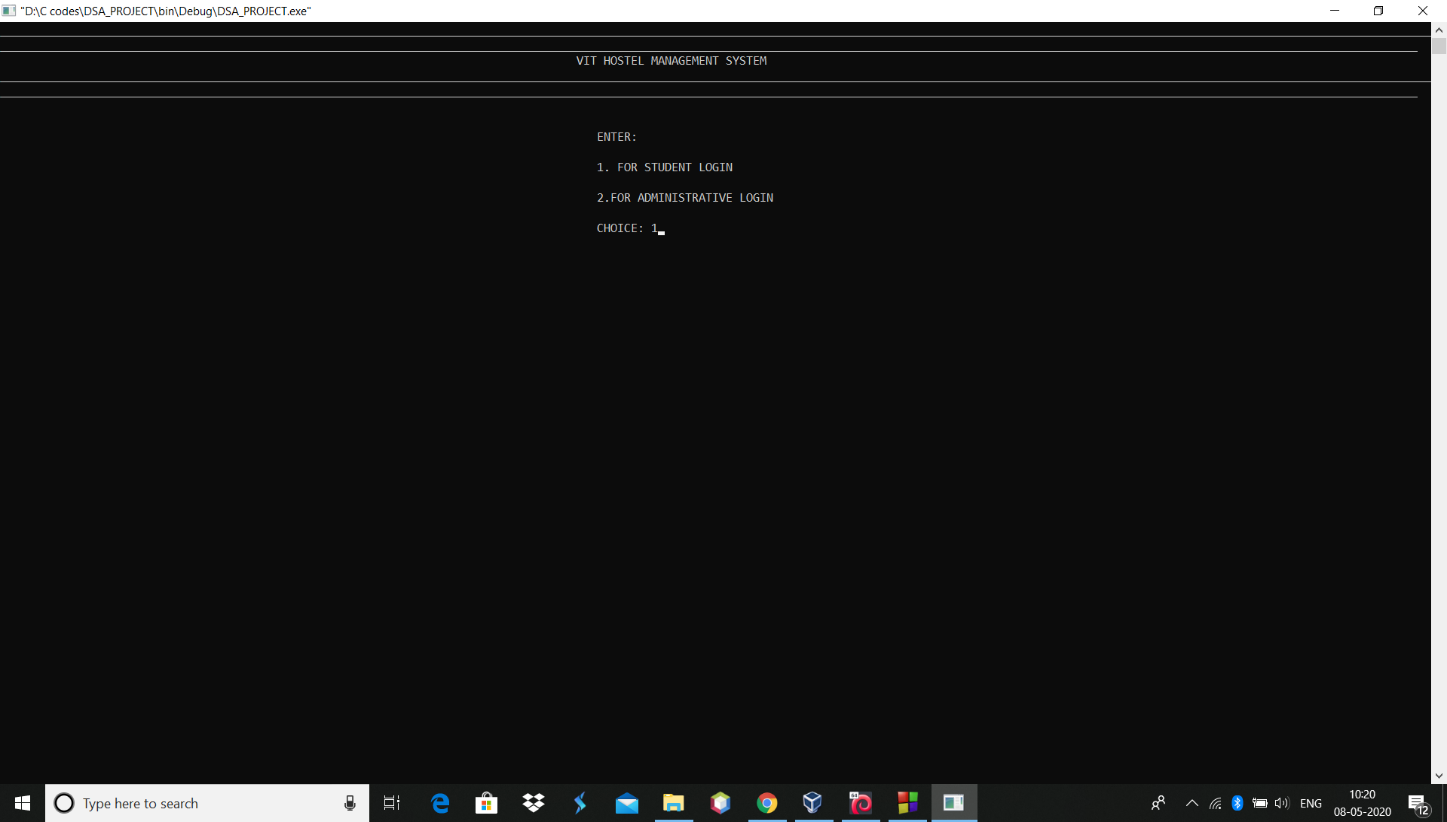
****

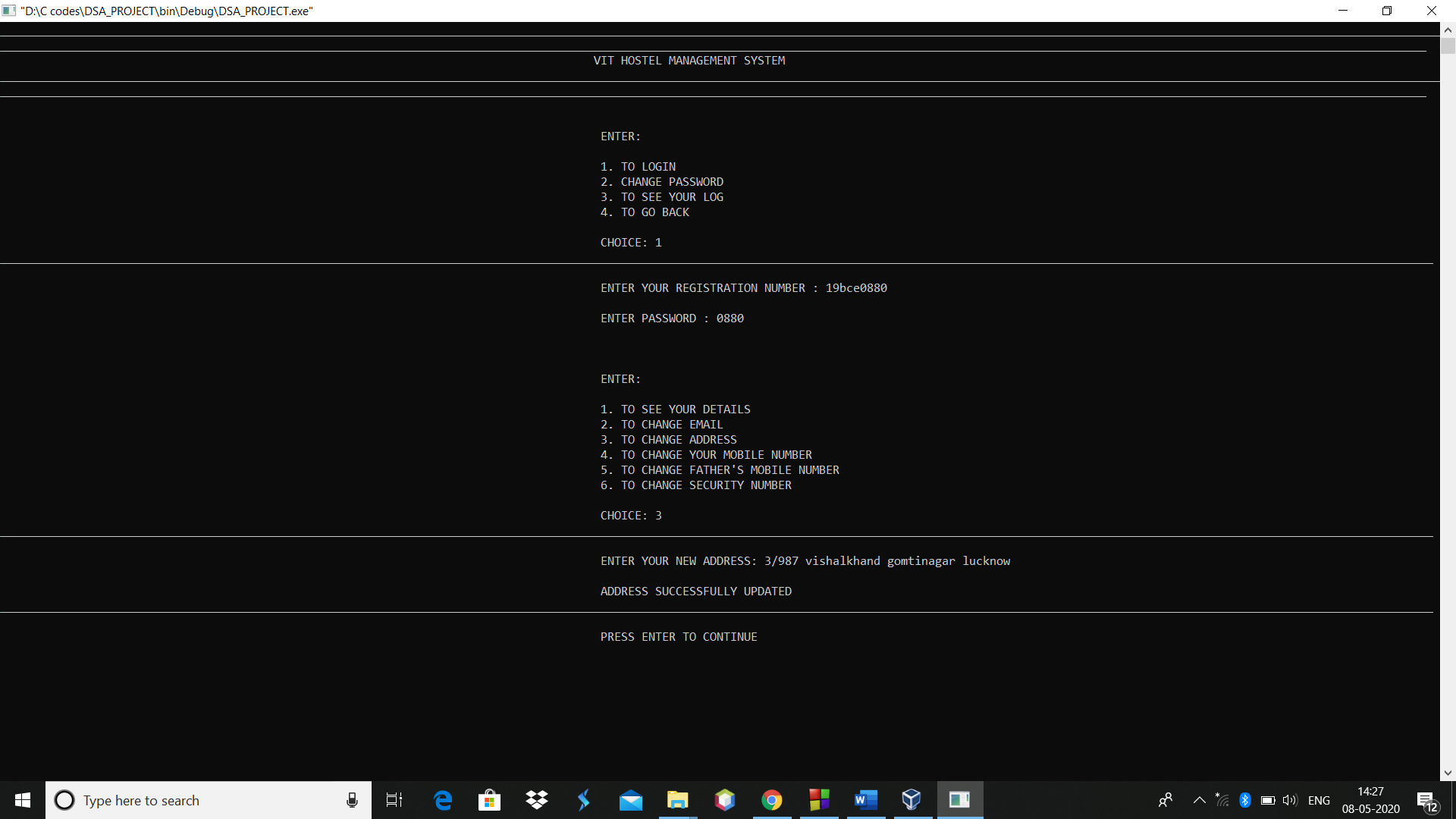
****

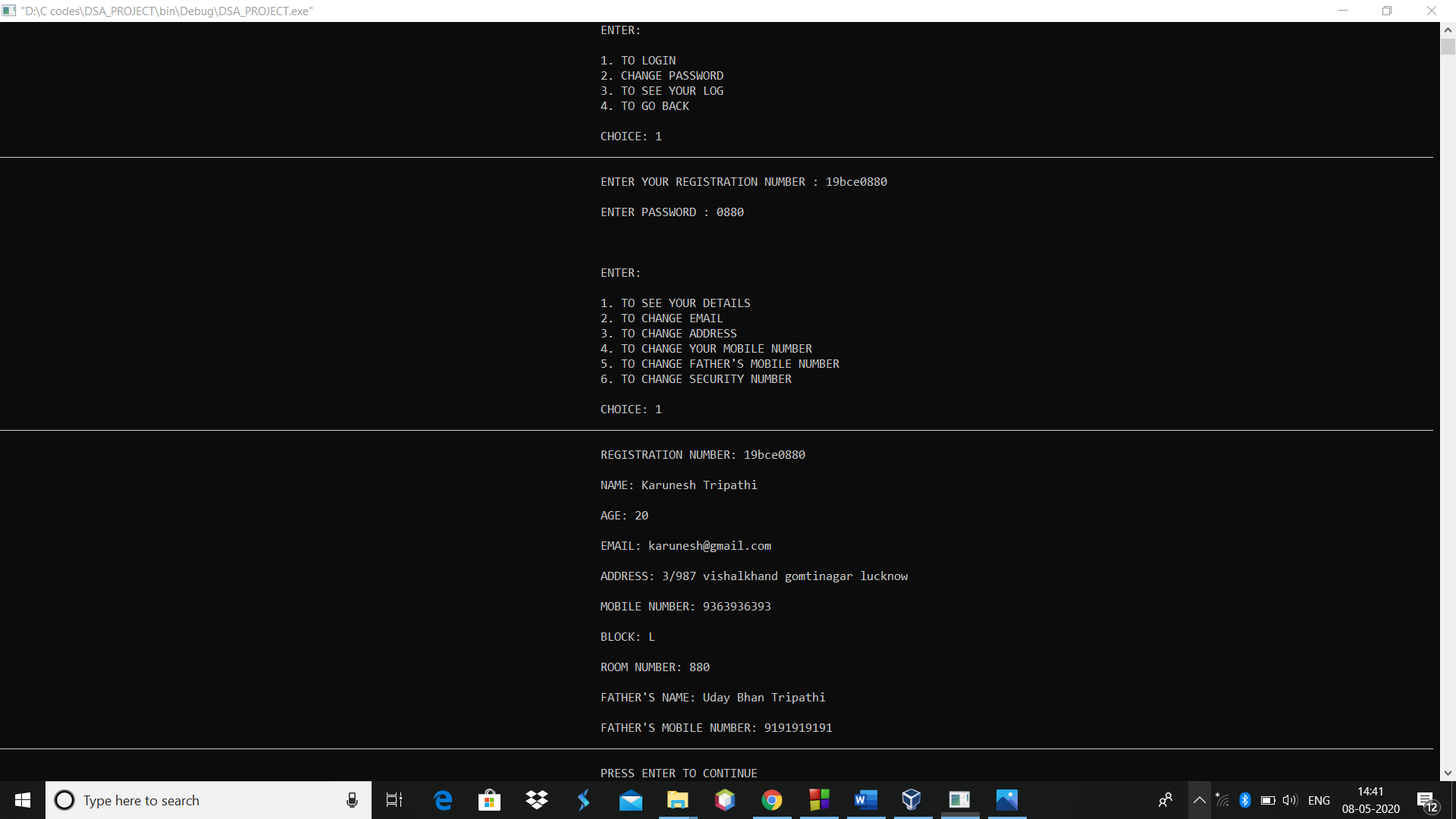
**TO CHANGE STUDENT’S EMAIL**

**  
**

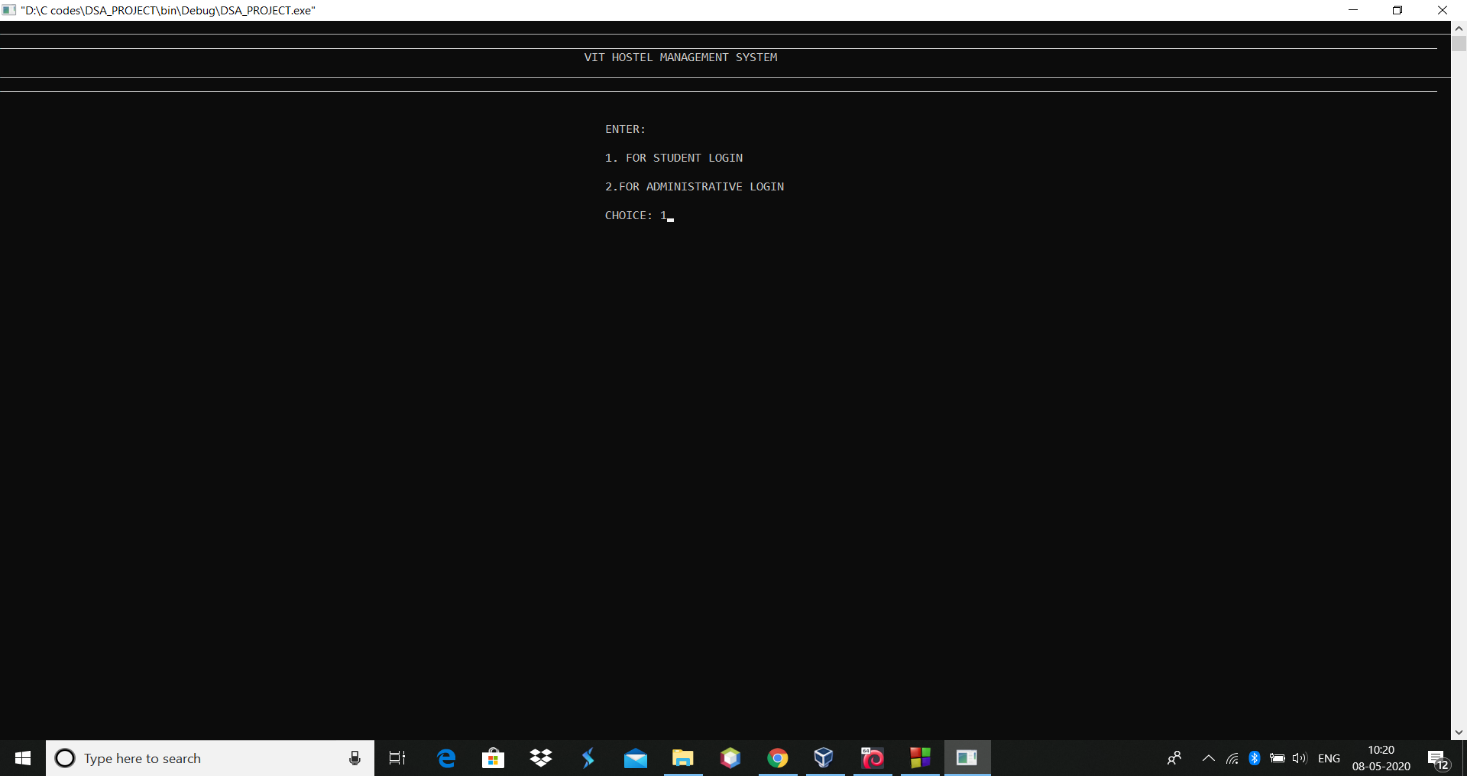
**TO CHANGE STUDENT’S ADDRESS**

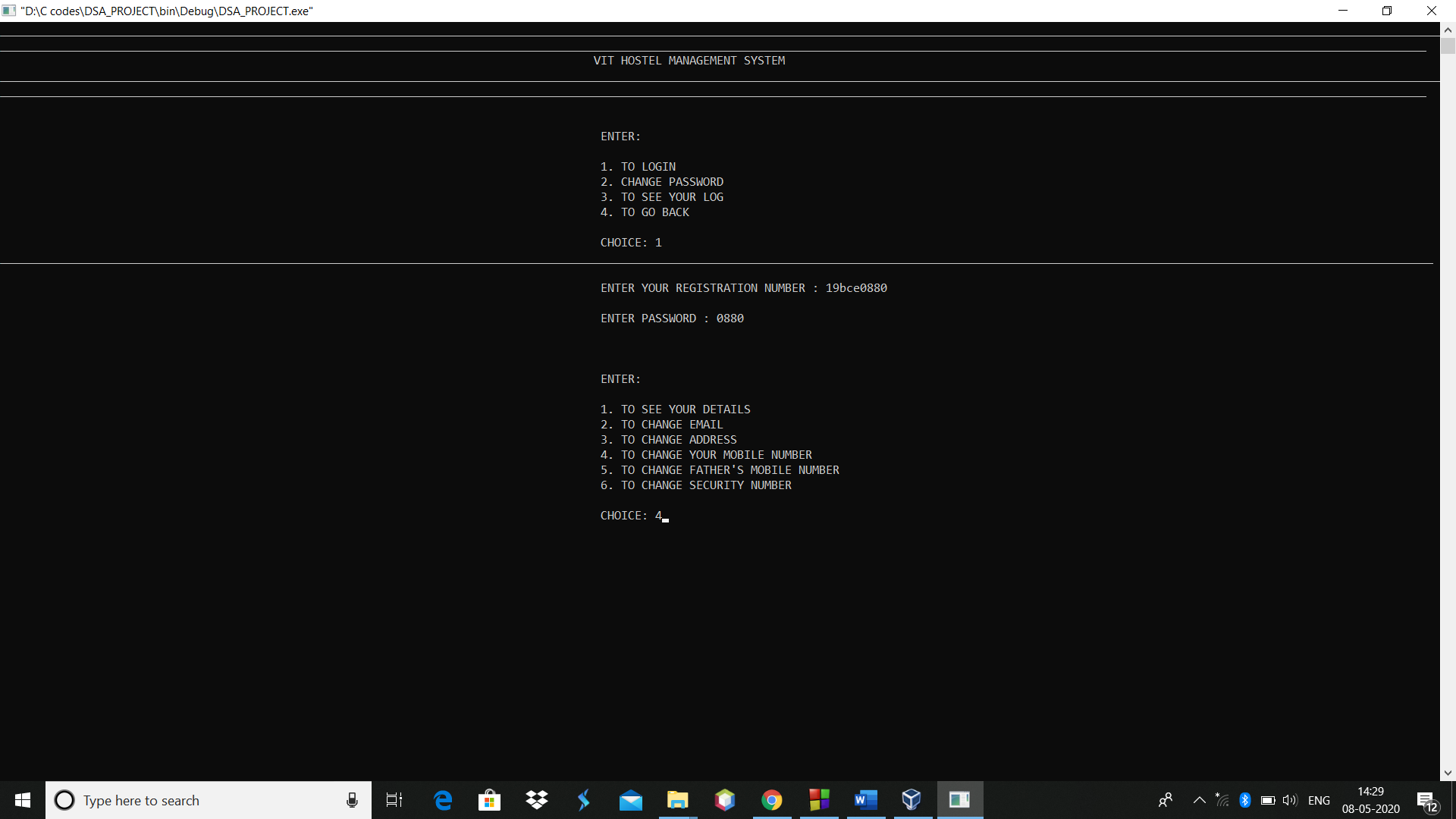
****

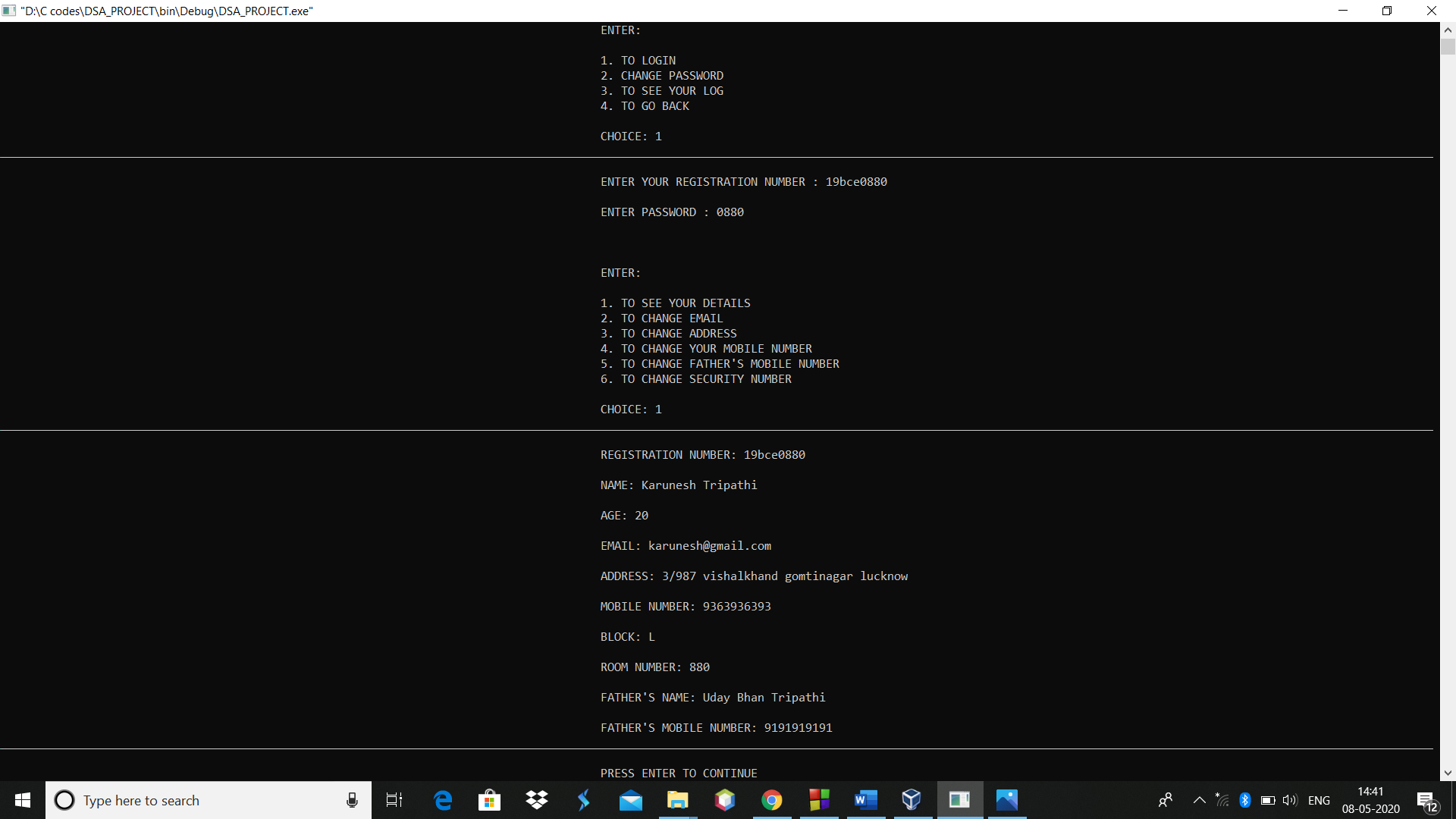
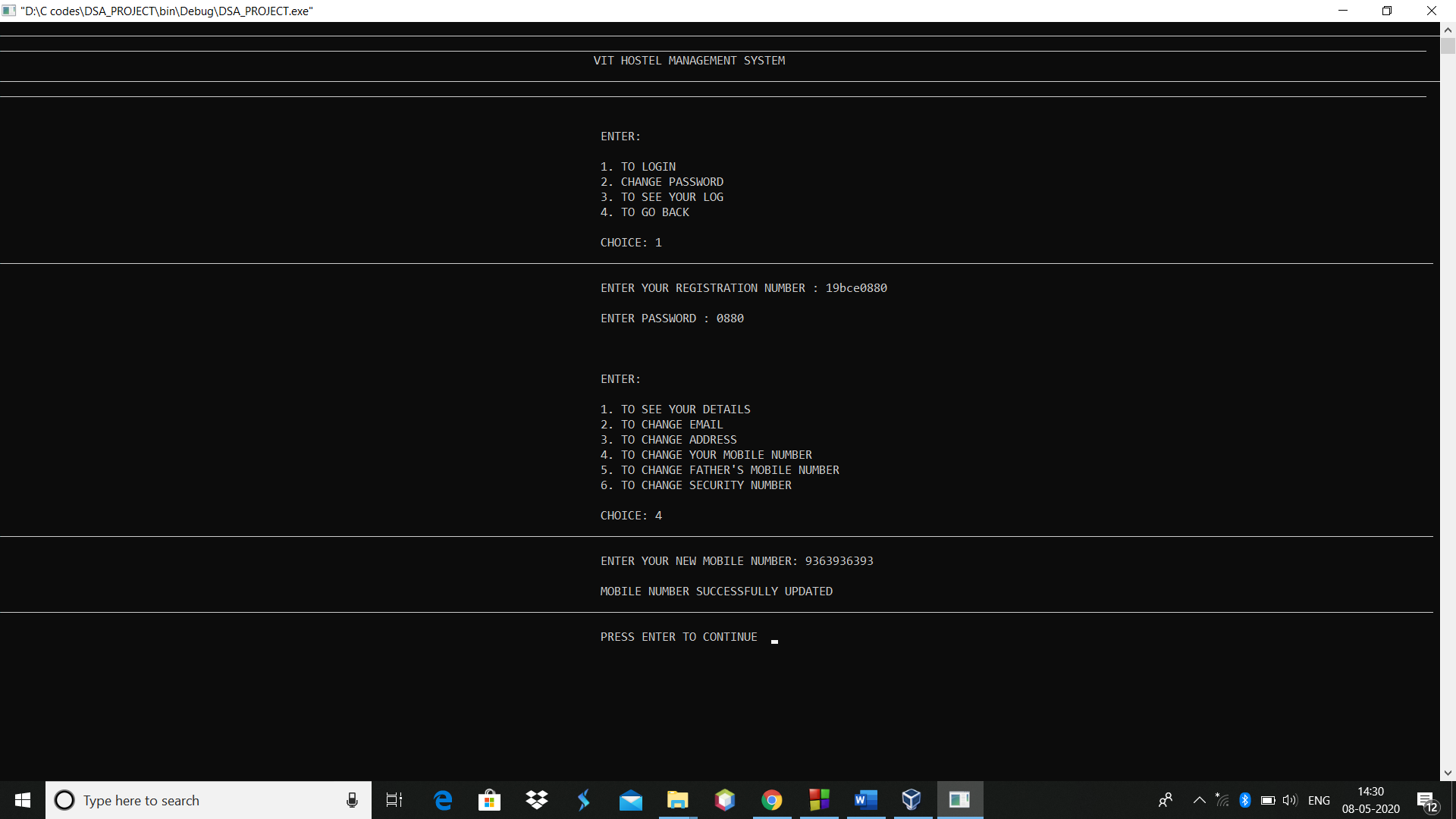
****

****

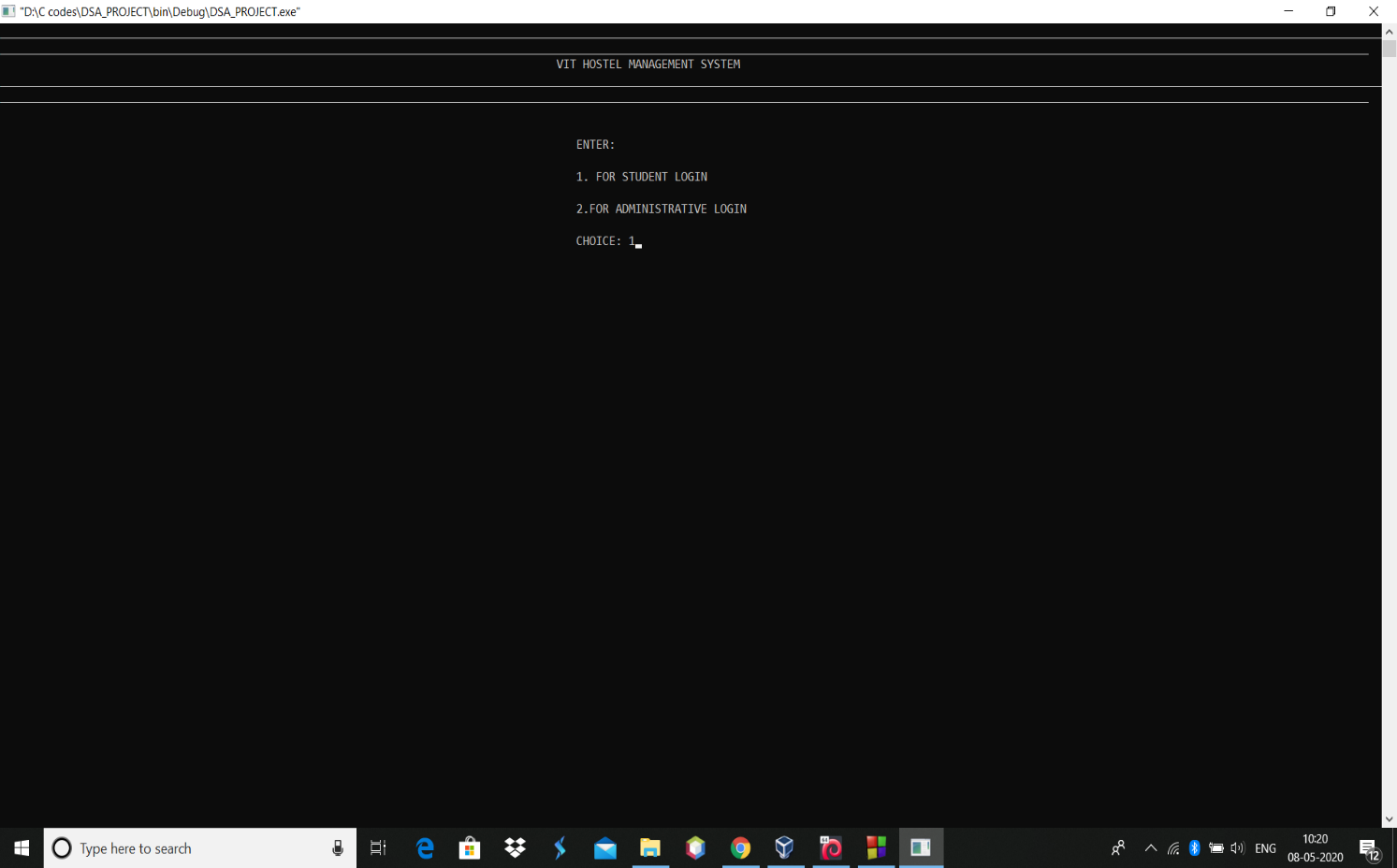
**TO CHANGE MOBILE NUMBER OF STUDENT**

****

****

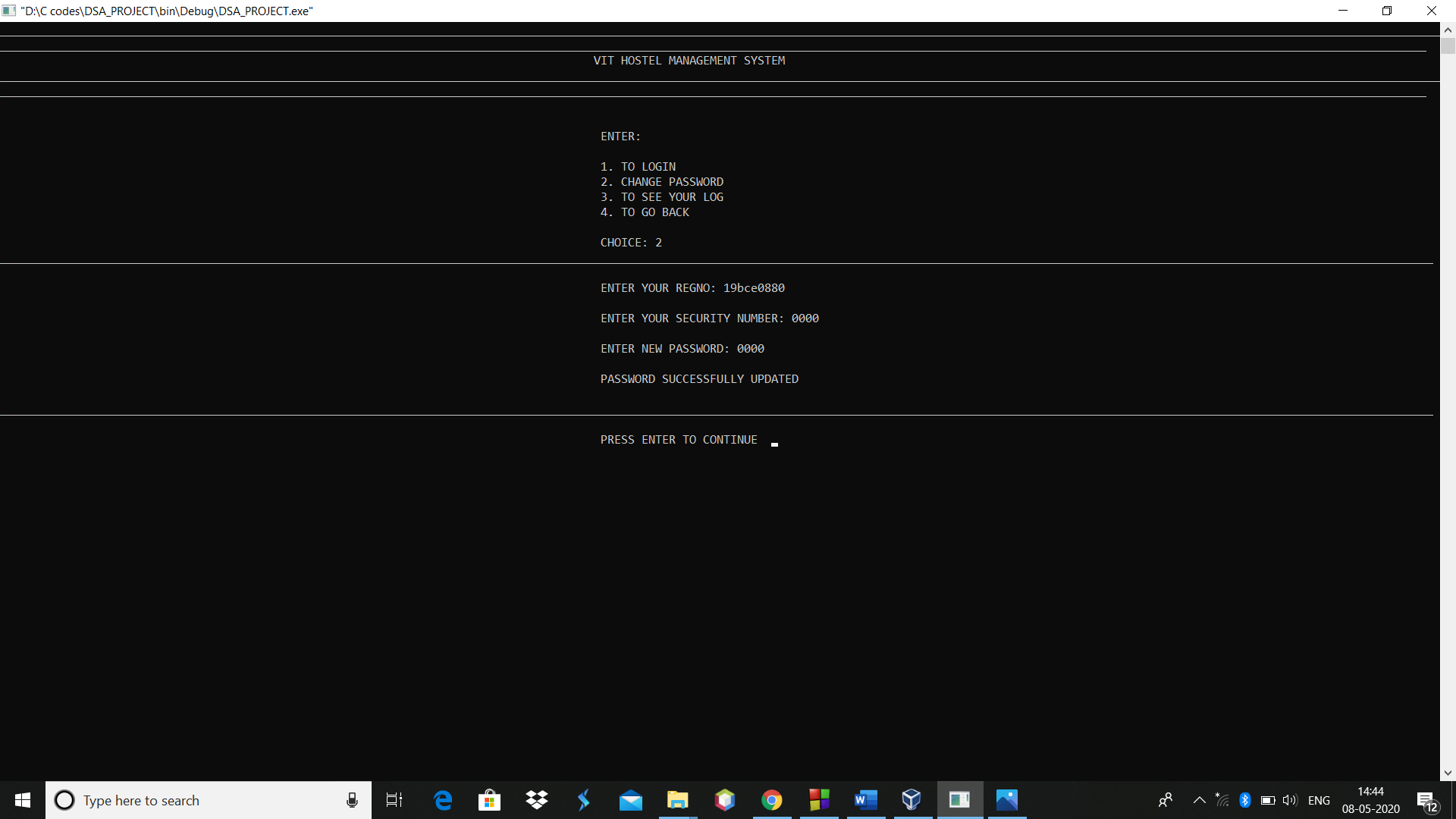
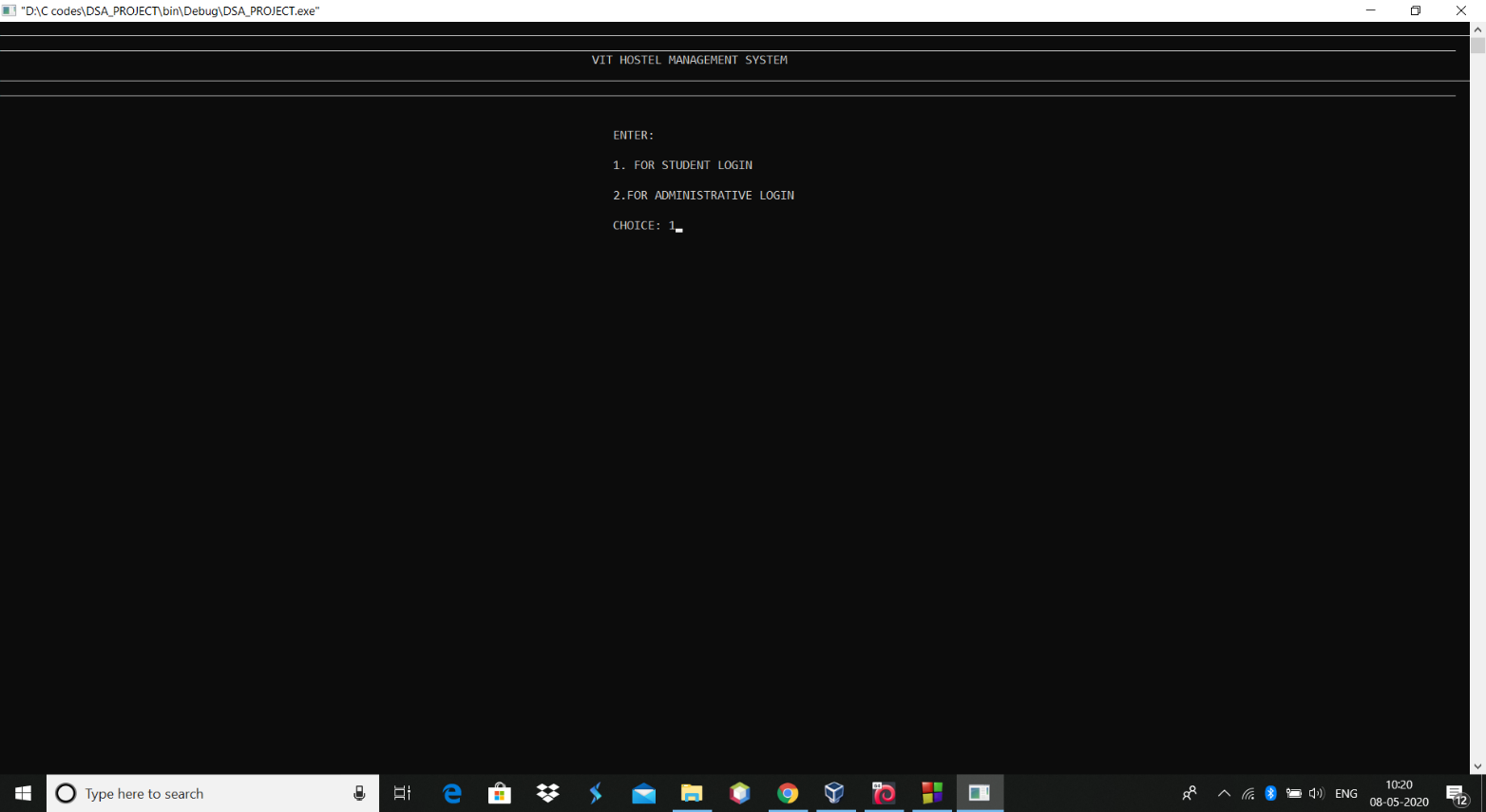
****

**TO CHANGE STUDENT’S FATHER’S MOBILE NUMBER**

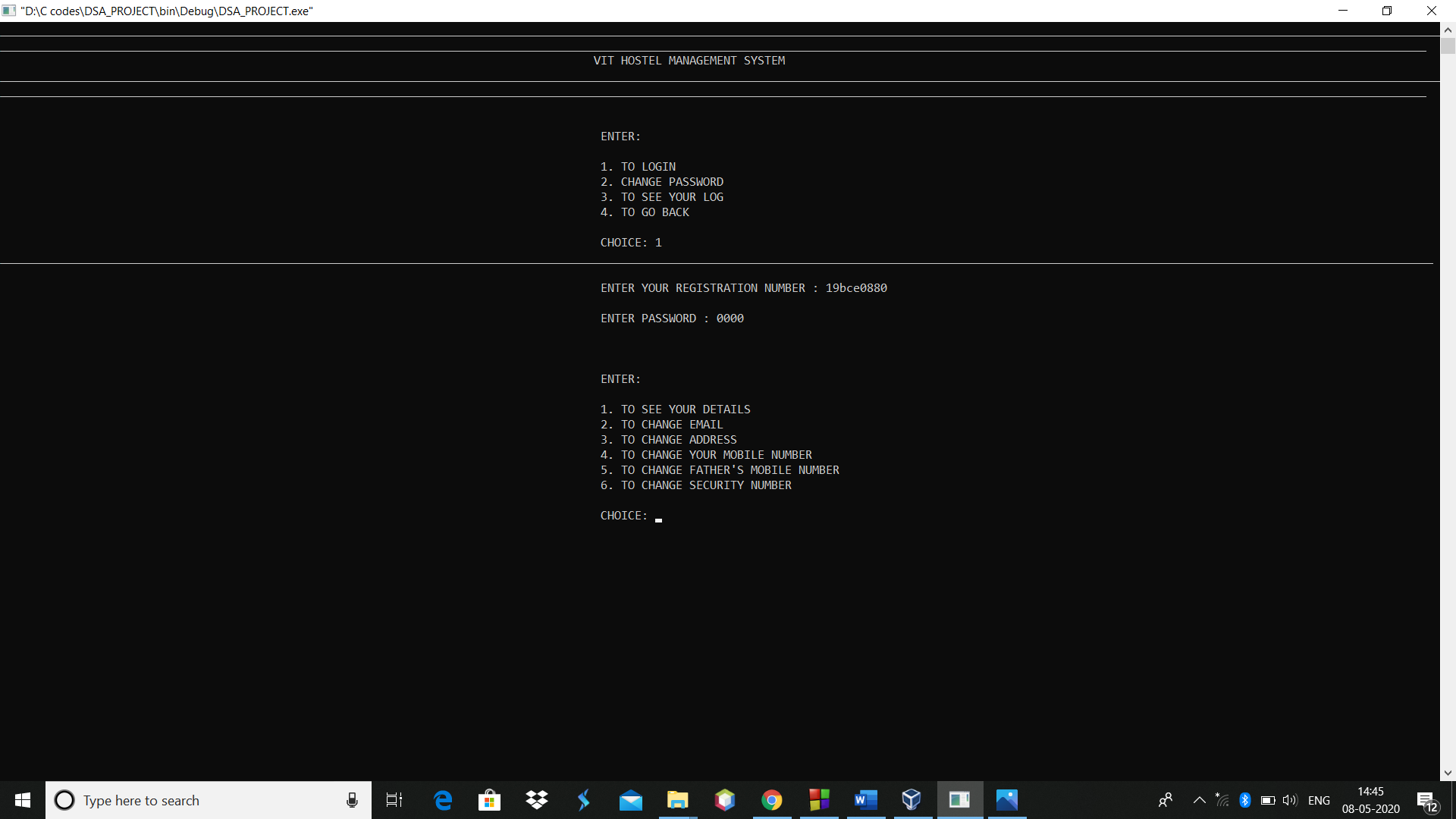
****

****

**TO CHANGE STUDENT’S SECURITY NUMBER**

****

**TO CHANGE PASSWORD USING UPDATED SECURITY NUMBER**

****