SPA PROJECT

# ENGINEERING ADMISSION ROUNDS AND STREAM ALLOCATION

## INTRODUCTION:

Following is the project that will assign streams to students based on their preferences for admission in a certain college.

It simulates the institutional rounds and automates the entire process based on the input of the students.

## WORKING:

1. The program takes input of data of students including their name, cet score, application number and preferences for streams. This input is read from a .csv file that already stores all the information.
2. The data file is linked with the program code and is read using File Handling.
3. The program then sorts all the students using Bubble Sort in a decreasing order on the basis of cet marks and correspondingly ranks them.
4. The students are then assigned streams based on their respective first preferences.
5. The program then asks user for logging in as either a student or a college personnel.
6. If the user is a student, then the program asks for the application number and displays the branch assigned to the student along with their application number and rank in a tabular form as the output.
7. If user selects college login, then the program asks the user to enter a password. If the entered password is correct, the program continues, else it terminates.
8. Once the user is authorised, application numbers of all students assigned to each branch and number of seats vacant in each branch are displayed. It also displays the application numbers of students not assigned any branch.
9. It then asks the user for the number of rounds to be conducted.
10. In case number of rounds is more than 1, the program asks each student if he is willing to continue to go to the next round.
11. If the branch the student is willing to pursue has availability, then the seat is assigned.
12. Once the assigning process is completed, the program again asks for a student or college login and the process is repeated till the number of rounds are completed.

MAIN CONCEPTS UTILIZED:

Array of structures

Bubble sorting of structure array

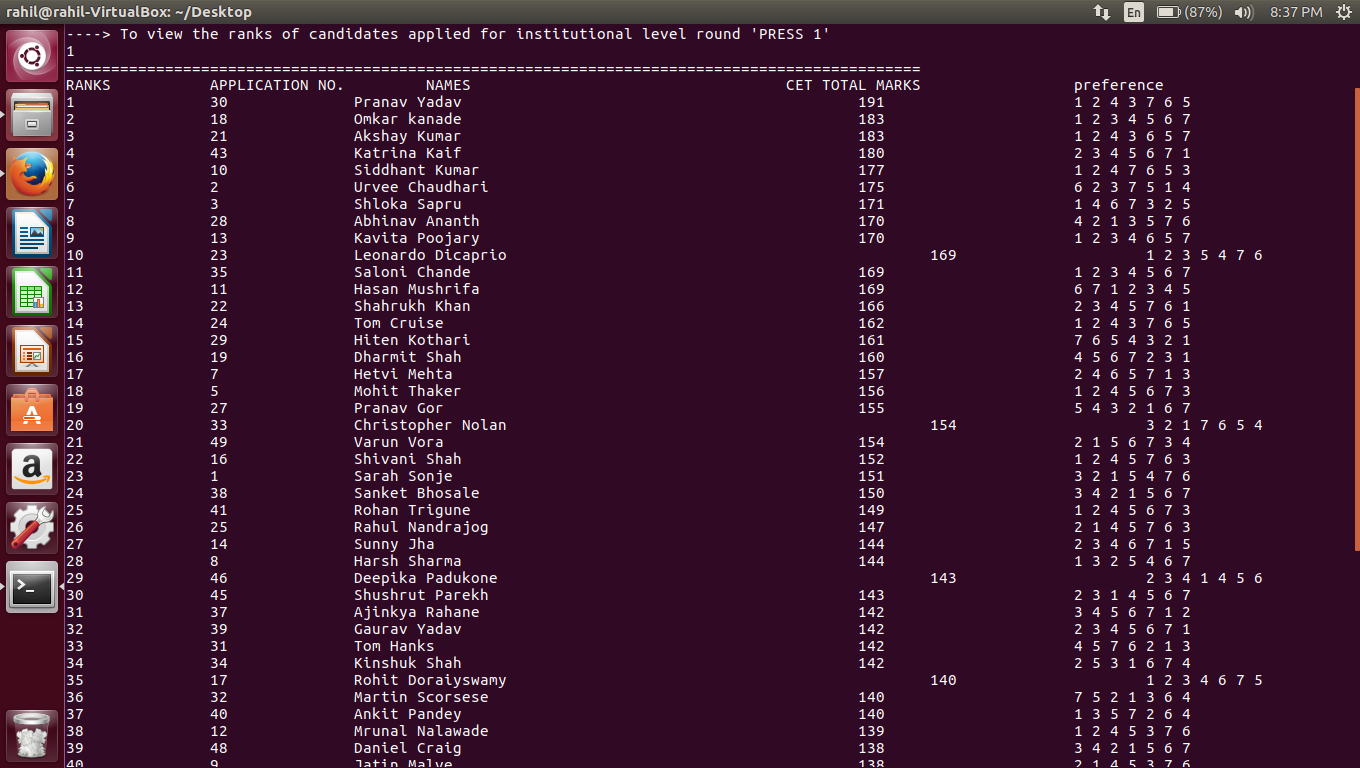
File handling: Taking Input from a Excel File

Converting the excel file to .csv format (comma separated values) and then use file handling.

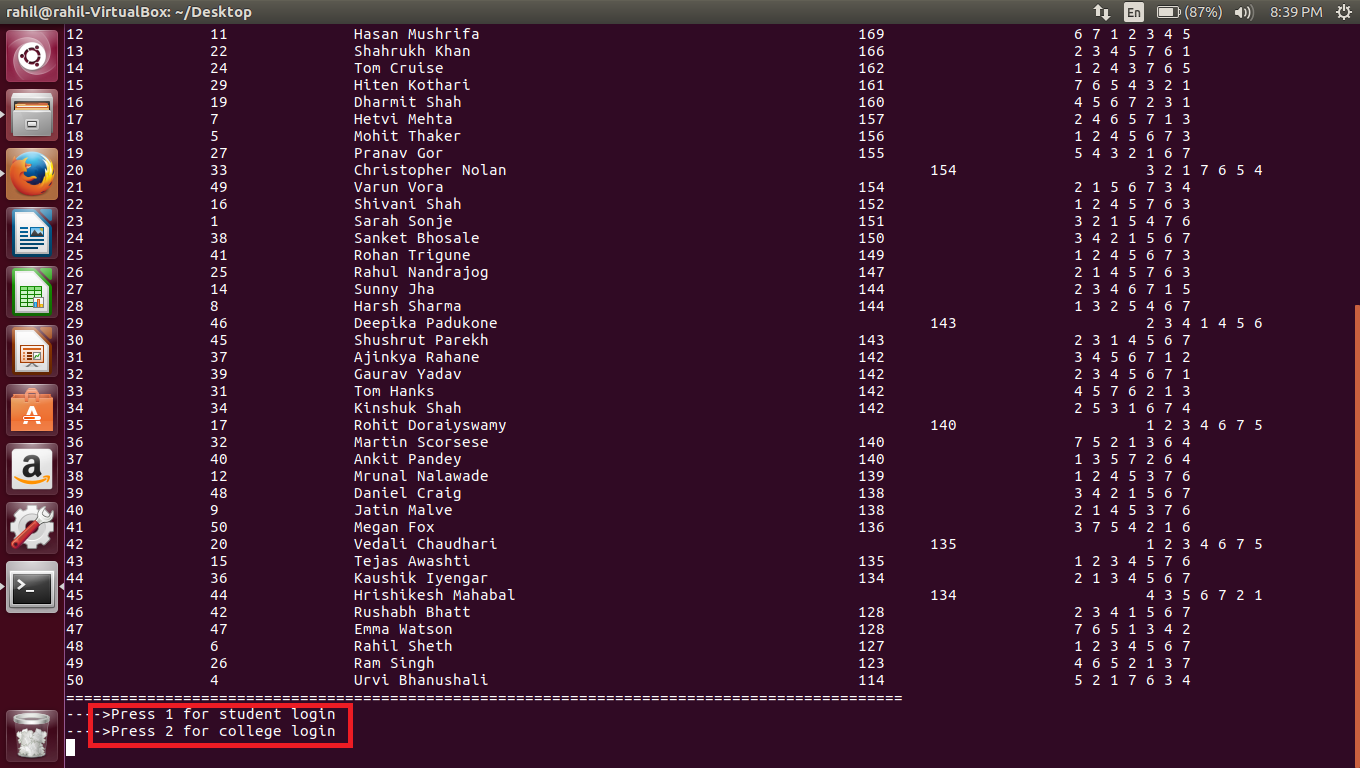
Authentication of the user

String comparison using the standard library of string.h

C Program Constructs used: If-else condition, for loops, switch case, call to functions



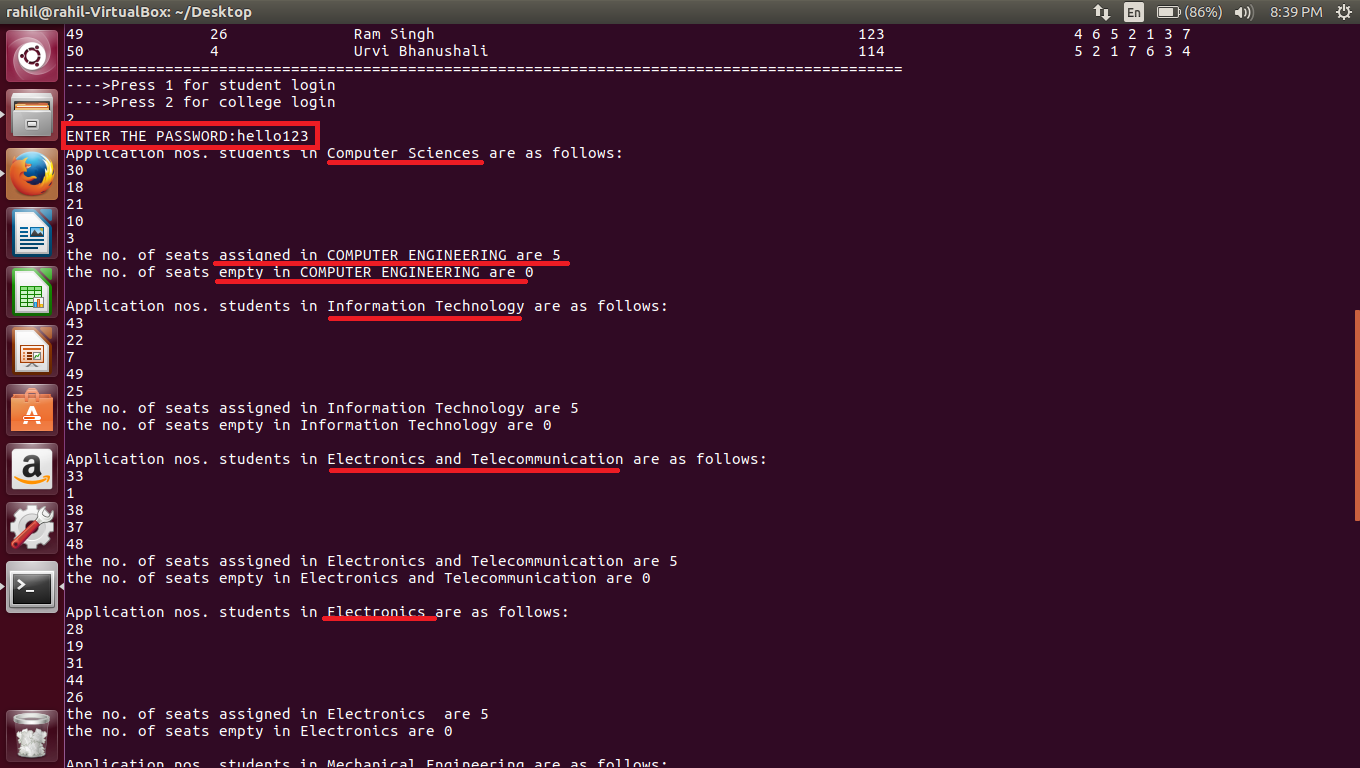
Shown above is the data of all students including their application number and preferences for each stream. The applicants are arranged in descending order on the basis of CET TOTAL MARKS.



The highlighted portion asks for user input as college or student.

**COLLEGE LOGIN**

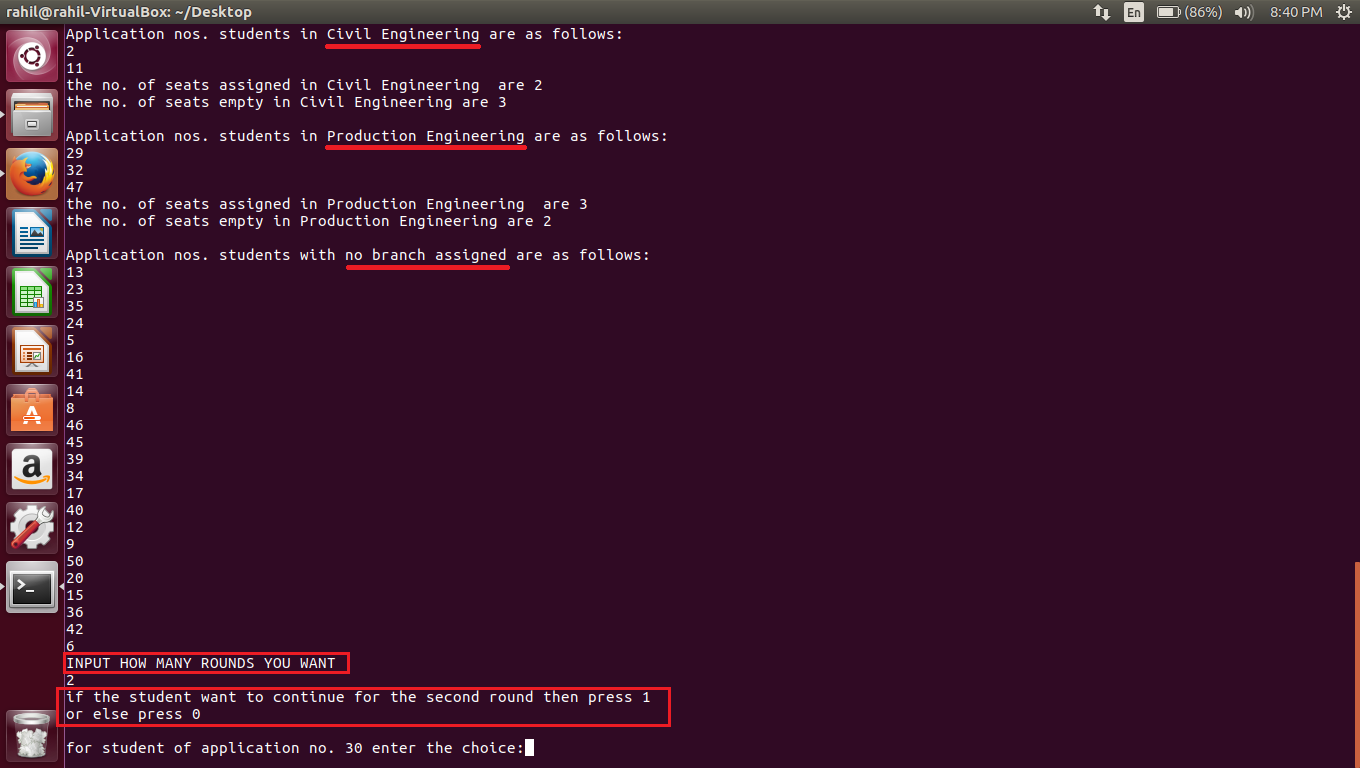
In case of selecting college login, the program asks for a password for authentication. The box in red indicates the password:



Once the password is entered and accepted, the program lists the applicants in each stream, and also displays the number of seats assigned and number of seats empty.

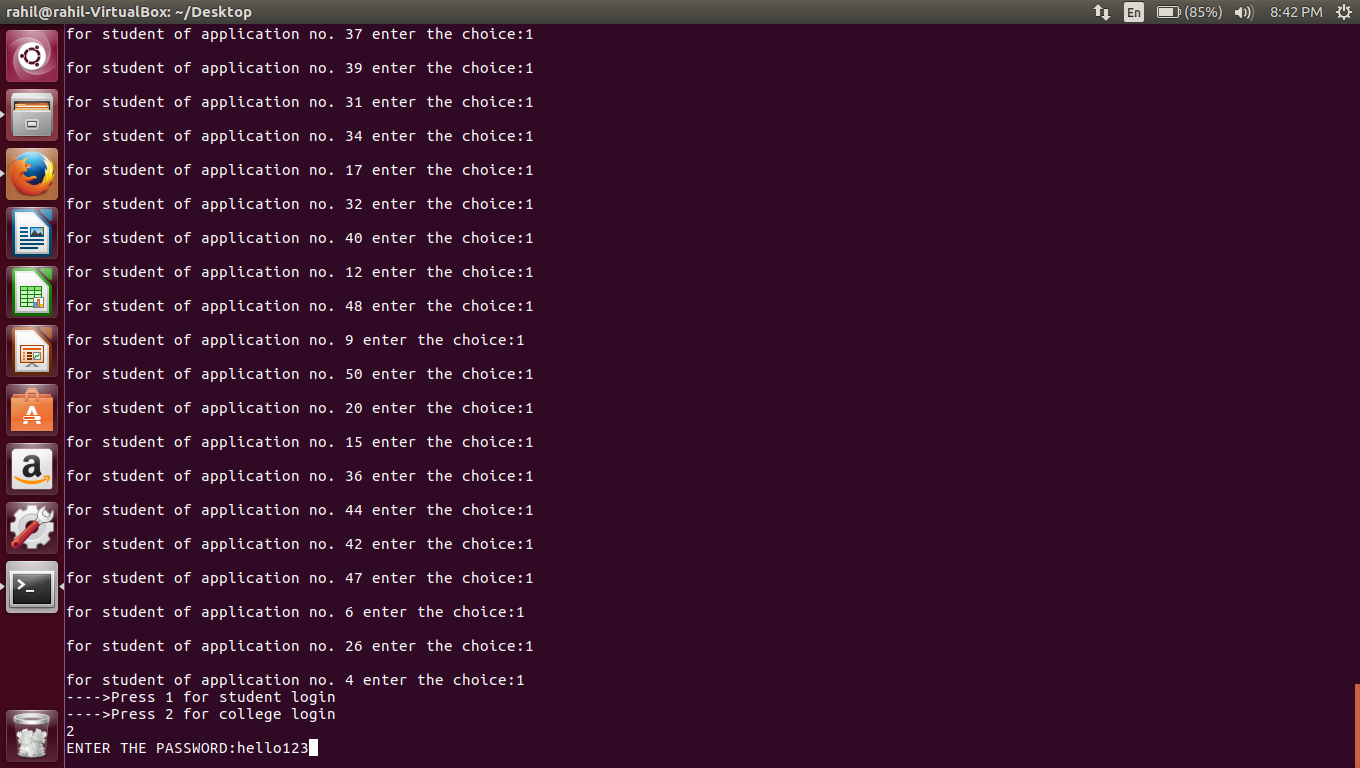
The streams allocated are Computer Engineering, Information Technology, Electronics and Telecommunications, Electronics, Mechanical Engineering, Civil Engineering and Production Engineering.

As the students are assigned to their branches based only on their first preferences, those students not assigned any branch are listed under a separate category. It displays the application numbers of those students who have not been assigned any branch after the first round.



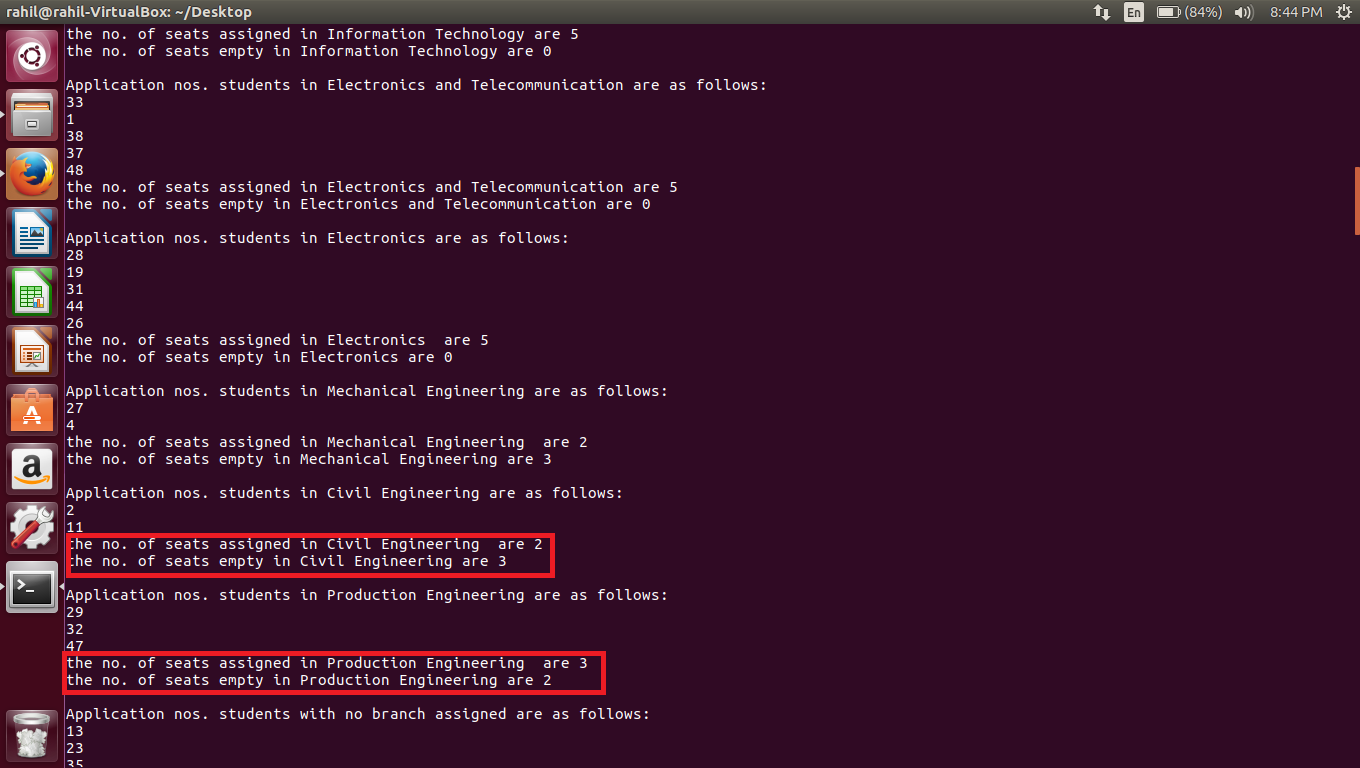
The user is then asked for the number of rounds they prefer.

The program then asks each student if they want to continue with the allotment in second round, or are satisfied with their current branch assignment.



Once inputs from all students are taken, the user is again asked for student or college login. Now if college login is selected, following changes are observed:

**BEFORE SECOND ROUND:**



Number of seats in civil engineering before second round are as follows:

Seats assigned: 2

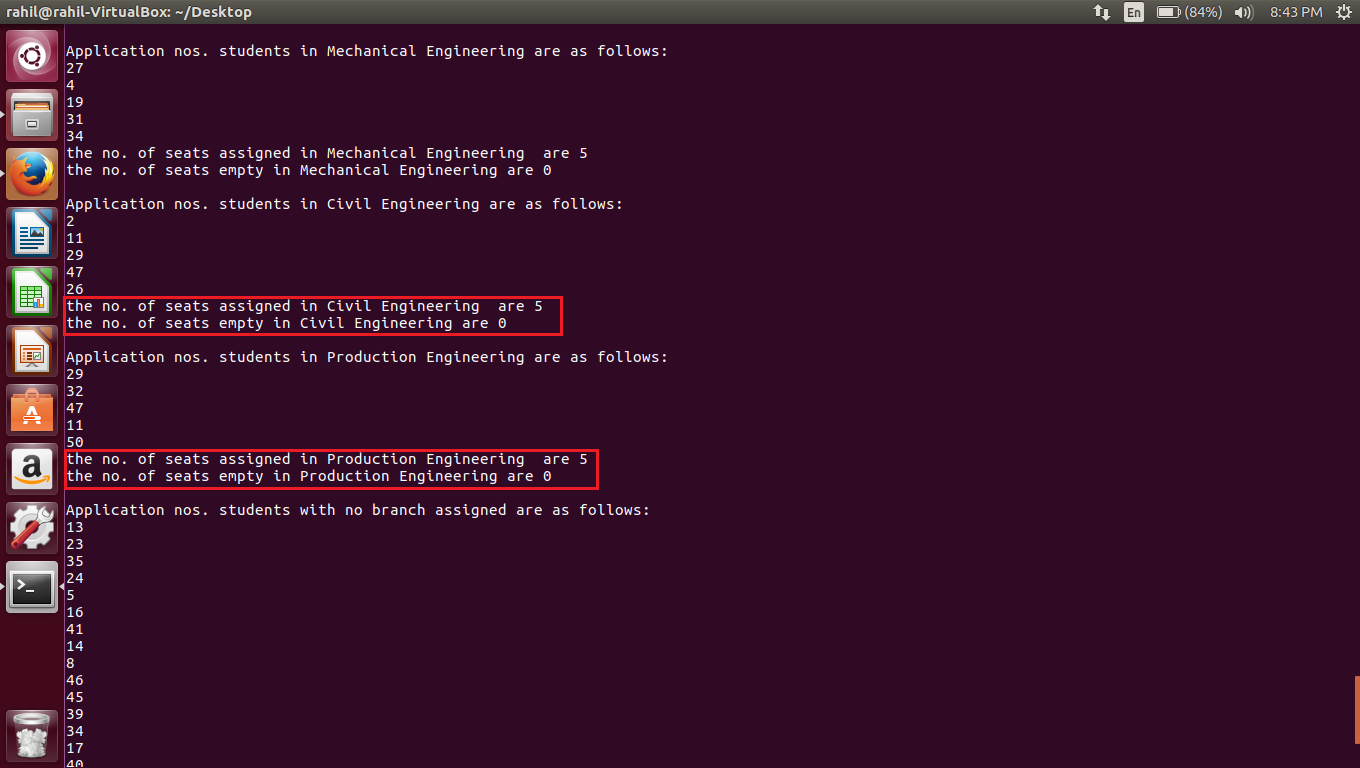
Seats Vacant: 3

Number of seats in Production Engineering before second round are:

Seats assigned: 3

Seats Vacant: 2

**AFTER SECOND ROUND:**



In civil engineering, number of seats assigned now are 5 and number of seats vacant are 0

In Production Engineering, number of seats assigned now are 5 and number of seats vacant are 0.

**CONCLUSION**:

Using the above program we can successfully conduct the admission process of any college and assign students the right stream on the basis of their merit and preferences.

**GROUP MEMBERS:**

* Shloka Sapru
* Rahil Sheth
* Sarah Sonje
* Mohit Thaker

**PROGRAM CODE**

#include <stdio.h>

#include <stdlib.h>

#include<string.h> //no of branches is 4

#define n 50 //no. of students

#define m 7 //no. of preferences

#define seat 5 //no of seats per branch

void swap(int,int);

void assign(int,int);

void print\_result(int);

void display(int);

struct student {

int appl\_no;

char names[50];

int math;

int phy;

int chem;

int cet;

int pref[m];

int choice;

//columns till m

}s[n];

static int i=0,k=0,j=0,f=0,g=0,p=0,q=0,r=0,t=0,u=0,nb=0;

int comps[seat][2], it[seat][2], extc[seat][2], etrx[seat][2],mech[seat][2],civil[seat][2],prod[seat][2], nobranch[n][2];

void main()

{

FILE\* fp;

fp=fopen("spadata.csv","r");

for(i=0;i<n;i++)

{

fscanf(fp,"%d, %[^,], %d,%d,%d,",&s[i].appl\_no, s[i].names, &s[i].math, &s[i].phy, &s[i].chem);

for(k=0;k<m-1;k++)

fscanf(fp,"%d,",&s[i].pref[k]);

fscanf(fp,"%d\n",&s[i].pref[k]);

s[i].cet = s[i].math + s[i].phy + s[i].chem;

}

fclose(fp);

for(i=0;i<n;i++)

{ for(k=i+1;k<n;k++)

{ if(s[i].cet<s[k].cet)

swap(i,k);

else if(s[i].cet==s[k].cet)

{ if(s[i].math<s[k].math)

swap(i,k);

else if(s[i].math==s[k].math)

{ if(s[i].phy<s[k].phy)

swap(i,k);

else if(s[i].phy==s[k].phy)

{ if(s[k].chem<s[k].chem)

swap(i,k);

}

}

}

}

}

printf("\t\t\*\*\*\*\*WELCOME TO INSTITUTIONAL LEVEL ROUNDS CONDUCTED BY S.P.I.T COLLEGE\*\*\*\*\*\n");

printf("\t\t\t\tTHANKS FOR APPLYING IN OUR INSTITUTE\n\n\n\n");

printf("----> To view the ranks of candidates applied for institutional level round 'PRESS 1'\n");

int q;

scanf("%d",&q);

if (q==1)

{

printf("===============================================================================================\n");

printf("RANKS\t\tAPPLICATION NO.\t\tNAMES\t\t\t\t\tCET TOTAL MARKS\t\t\tpreference\n");

for(i=0;i<n;i++)

{ printf("%d\t\t%d\t\t%s\t\t\t\t\t\t%d\t\t\t",i+1, s[i].appl\_no, s[i].names, s[i].cet);

for(k=0;k<m;k++)

{ printf("%d ",s[i].pref[k]);

}

printf("\n");

}

printf("=============================================================================================\n");}

for(i=0;i<n;i++)

{

assign(i,0);

}

display(1);

printf("INPUT HOW MANY ROUNDS YOU WANT\n");

int round,tp;

scanf("%d",&round);

printf("If the student want to continue for the next round then press 1\nOR\nIf you accept the assisgned branch press 0\n");

for(tp=1;tp<round;tp++)

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

{ printf("\nFor student of application no. %d enter the choice:",s[i].appl\_no);

scanf("%d",&s[i].choice);

}

for (i=0;i<n;i++)

{ if(s[i].choice==1)

{ assign(i,tp);

}

}

display(1);

}

}

void swap(int i, int k)

{

struct student temp;

temp=s[i];

s[i]=s[k];

s[k]=temp;

}

void assign(int i,int j)

{ { if(s[i].pref[j]==1 && f<seat)

{ comps[f][1]=s[i].appl\_no;

comps[f][2]=i+1;

f++;

return;

}

else if(s[i].pref[j]==2 && g<seat)

{ it[g][1]=s[i].appl\_no;

it[g][2]=i+1;

g++;

return;

}

else if(s[i].pref[j]==3 && p<seat)

{ extc[p][1]=s[i].appl\_no;

extc[p][2]=i+1;

p++;

return;

}

else if(s[i].pref[j]==4 && q<seat)

{ etrx[q][1]=s[i].appl\_no;

etrx[q][2]=i+1;

q++;

return;

}

else if(s[i].pref[j]==5 && r<seat)

{ mech[r][1]=s[i].appl\_no;

mech[r][2]=i+1;

r++;

return;

}

else if(s[i].pref[j]==6 && t<seat)

{ civil[t][1]=s[i].appl\_no;

civil[t][2]=i+1;

t++;

return;

}

else if(s[i].pref[j]==7 && u<seat)

{ prod[u][1]=s[i].appl\_no;

prod[u][2]=i+1;

u++;

return;

}

else if(j==0)

{ nobranch[nb][1]=s[i].appl\_no;

nobranch[nb][2]=i+1;

nb++;

return;

}

}

}

void print\_result(int v)

{

printf("Enter your application no to check your allotment\n");

scanf("%d",&v);

for(i=0;i<n;i++)

{ if(v==comps[i][1]&&i<f)

{ printf("\t\t!!!CONGRATULATIONS!!!\n");

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

printf("RANK\t\t\tAPPLICATION NO.\t\t\tSEAT ALLOTED IN BRANCH\n");

printf("%d\t\t\t\t%d\t\t\t",comps[i][2],comps[i][1]);

printf("COMPUTER ENGINEERING\n");

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

exit(1);

}

else if(v==it[i][1] && i<g)

{ printf("\t\t!!!CONGRATULATIONS!!!\n");

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

printf("RANK\t\t\tAPPLICATION NO.\t\t\tSEAT ALLOTED IN BRANCH\n");

printf("%d\t\t\t\t%d\t\t\t",it[i][2],it[i][1]);

printf("INFORMATION TECHNOLOGY\n");

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

exit(1);

}

else if(v==extc[i][1] && i<p)

{ printf("\t\t!!!CONGRATULATIONS!!!\n");

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

printf("RANK\t\t\tAPPLICATION NO.\t\t\tSEAT ALLOTED IN BRANCH\n");

printf("%d\t\t\t\t%d\t\t\t",extc[i][2],extc[i][1]);

printf("ELECTRONICS AND TELECOMMUNICATION\n");

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

exit(1);

}

else if(v==etrx[i][1] && i<q)

{ printf("\t\t!!!CONGRATULATIONS!!!\n");

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

printf("RANK\t\t\tAPPLICATION NO.\t\t\tSEAT ALLOTED IN BRANCH\n");

printf("%d\t\t\t\t%d\t\t\t",etrx[i][2],etrx[i][1]);

printf("ELECTRONICS\n");

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

exit(1);

}

else if(v==mech[i][1] && i<q)

{ printf("\t\t!!!CONGRATULATIONS!!!\n");

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

printf("RANK\t\t\tAPPLICATION NO.\t\t\tSEAT ALLOTED IN BRANCH\n");

printf("%d\t\t\t\t%d\t\t\t",mech[i][2],mech[i][1]);

printf("MECHANICAL ENGINEERING\n");

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

exit(1);

}

else if(v==civil[i][1] && i<q)

{ printf("\t\t!!!CONGRATULATIONS!!!\n");

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

printf("RANK\t\t\tAPPLICATION NO.\t\t\tSEAT ALLOTED IN BRANCH\n");

printf("%d\t\t\t\t%d\t\t\t",civil[i][2],civil[i][1]);

printf("CIVIL ENGINEERING\n");

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

exit(1);

}

else if(v==prod[i][1] && i<q)

{ printf("\t\t!!!CONGRATULATIONS!!!\n");

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

printf("RANK\t\t\tAPPLICATION NO.\t\t\tSEAT ALLOTED IN BRANCH\n");

printf("%d\t\t\t\t%d\t\t\t",prod[i][2],prod[i][1]);

printf("PRODUCTION ENGINEERING\n");

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

exit(1);

}

else if(v==nobranch[i][1] && i<nb)

{ printf("Your rank is %d\n", nobranch[i][2]);

printf("You havent been alloted any seat\n");

exit(1);

}

}

}

void display(int w)

{int z,v=1;

printf("---->Press 1 for student login\n---->Press 2 for college login \n");

scanf("%d",&z);

if (z==1)

{ print\_result(v);

}

if(z==2)

{ char passwrd[50]="hello123",pass[50];

printf("ENTER THE PASSWORD:");

scanf("%s",pass);

if(strcmp(passwrd,pass)!=0)

{ printf("\*\*Invalid password\*\*\nPlease try Again\n");

}

else

{ printf("Application nos. students in COMPUTER ENGINEERING are as follows:\n");

for(i=0;i<f;i++)

{ printf("%d\n",comps[i][1]);

}

printf("No. of seats assigned: %d\nNo. of seats empty: %d\n\n",f,seat-f);

printf("Application nos. students in INFORMATION TECHNOLOGY are as follows:\n");

for(i=0;i<g;i++)

{ printf("%d\n",it[i][1]);

}

printf("No. of seats assigned: %d\nNo. of seats empty: %d\n\n",g,seat-g);

printf("Application nos. students in ELECTRONICS AND TELECOMMUNICATION are as follows:\n");

for(i=0;i<p;i++)

{ printf("%d\n",extc[i][1]);

}

printf("No. of seats assigned: %d\nNo. of seats empty: %d\n\n",p,seat-p);

printf("Application nos. students in ELECTRONICS are as follows:\n");

for(i=0;i<q;i++)

{ printf("%d\n",etrx[i][1]);

}

printf("No. of seats assigned: %d\nNo. of seats empty: %d\n\n",q,seat-q);

printf("Application nos. students in MECHANICAL ENGINEERING are as follows:\n");

for(i=0;i<r;i++)

{ printf("%d\n",mech[i][1]);

}

printf("No. of seats assigned: %d\nNo. of seats empty: %d\n\n",r,seat-r);

printf("Application nos. students in CIVIL ENGINEERING are as follows:\n");

for(i=0;i<t;i++)

{ printf("%d\n",civil[i][1]);

}

printf("No. of seats assigned: %d\nNo. of seats empty: %d\n\n",t,seat-t);

printf("Application nos. students in PRODUCTION ENGINEERING are as follows:\n");

for(i=0;i<u;i++)

{ printf("%d\n",prod[i][1]);

}

printf("No. of seats assigned: %d\nNo. of seats empty: %d\n\n",u,seat-u);

printf("Application nos. students with no branch assigned are as follows:\n");

for(i=0;i<nb;i++)

{ printf("%d\n",nobranch[i][1]);

}

}

}

}