**Lab Assignment 3**

**(Week 2 – Lab A)**

**Q1.** You have been given the snippet of a C++ program. Three classes named as subject, student, and faculty are there in the given program. Identify the relationship among these classes and draw the class diagram.

class subject

{

int sub\_Id;

char sub\_Name[20];

int sub\_SemesterOffered;

int sub\_BranchOfferedBy;

public:

subject()

{

cin>>sub\_Id;

cin>>sub\_Name;

cin>>sub\_SemesterOffered;

cin>>sub\_BranchOfferedBy;

}

char \* retSubName();

int retSub(char \*tempName);

};

int subject::retSub(char \*tempName)

{

if(!strcmp(sub\_Name, tempName))

return 1;

else

return 0;

}

char \* subject::retSubName()

{

return sub\_Name;

}

class student

{

int std\_Id; char std\_Name[20]; int std\_Semester;

char std\_Branch[10]; subject \*registeredSub[5];

int subRegCount;

public:

student()

{

cin>>std\_Id;

cin>>std\_Name;

cin>>std\_Semester;

cin>>std\_Branch;

subRegCount = -1;

}

void subjectRegistration(subject \*temp);

int retRegSub(char \*tempName);

char \* retStdName();

};

void student::subjectRegistration(subject \*temp)

{

registeredSub[++subRegCount] = temp;

}

int student::retRegSub(char \*tempName)

{

int t;

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

{

if(!strcmp(registeredSub[t] -> retSubName(), tempName))

return 1;

}

return 0;

}

char \* student::retStdName() { return std\_Name; }

class faculty

{

int fac\_Id; char fac\_Name[20]; char fac\_Branch[10]; subject \*teachesSub[3]; int subTeachesCount;

public:

faculty()

{

cin>>fac\_Id;

cin>>fac\_Name[20];

cin>>fac\_Branch[10];

subTeachesCount = -1;

}

void subjectTeaches(subject \*temp);

};

void faculty::subjectTeaches(subject \*temp)

{

teachesSub[++subTeachesCount] = temp;

}

int main()

{

subject \*sub[20]; student \*std[1000]; faculty \*fac[10]; int i, j, k, count; char sName[20];

for(i = 0; i < subCount; i++) { sub[i] = new subject; }

for(i = 0; i < stdCount; i++) { std[i] = new student; }

for(i = 0; i < facCount; i++) { fac[i] = new faculty; }

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

{

cout<<"\nEnter subjects count registered by"<<i<<"th student"; cin>>count;

for(j = 0; j < count; j++)

{

cout<<"\nEnter name of "<<j<<"th subject to be registered"; cin>>sName;

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

{

if(sub[k] -> retSub(sName)) { std[i] -> subjectRegistration(sub[k]); break; }

}

}

}

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

{

cout<<"\nEnter subjects count taught by"<<i<<"th faculty"; cin>>count;

for(j = 0; j < count; j++)

{

cout<<"\nEnter name of "<<j<<"th subject to be taught";

cin>>sName;

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

{

if(sub[k] -> retSub(sName)) { fac[i] -> subjectTeaches(sub[k]); break; }

}

}

}

Besides the three classes, the main() function is given to you where database of 20, 1000, and 10 subjects, students, and faculty are created respectively. Further the relationships among these classes are also implemented. In this program, subCount, stdCount, and facCount are global variables and initials some values lesser than 20, 1000, and 10 respectively.

Implement necessary functions in these classes (subjects, students, and faculty) and append the main() function to perform following queries:

1. Print the name of student(s) who registered in user inputted subject (e.g. Data Structures)
2. Print the name of the subject(s) in which maximum number of student registered
3. Print the name of the faculty(s) who is teaching maximum number of students

**Q2.** Mobile phones are manufactured by many companies, e.g. Samsung, Sony, etc. Usually performance of these mobile phones are measured by memory (in GB), front camera (in MP), rear camera (in MP), and battery backup (in Hr.) and accordingly price of the mobile phones varies. To operate mobile phones, users need sim-card which is to be issued by service provider, e.g. Airtel, Vodafone, etc. Besides the storage capacity, each sim is uniquely identified by the Mobile Number. When a sim is placed into a mobile, then only a mobile is functional otherwise not. Users need to purchase mobile as well as sim to make calls, use internet, etc. In each such phone, user can store contact details of his/her friends in the mobile storage or sim storage. To make a call, user is required to search the mobile number of his/her friend from the contact list (either of phone or of sim or of both). Whenever a call is made from one user (say X) to another user (say Y), it is required to display the message ―Airtel/Vodafone/etc. user with mobile no. XXXXXX is calling on the mobile screen of Y. Record of this call is to be stored into both users mobile phone: for X, it is outgoing and for Y, it is incoming. Further, a user may have maximum 5 bank accounts, where he/she can perform the transactions (either deposit or withdrawal). Withdrawal is based on the available balance in an account of the user. Based on the available balance in all accounts of a user, he/she decides whether a specific mobile can be purchased or not. Draw the class diagram and implement it in C++ and answer following:

(a) Who are the users capable enough to purchase a specific mobile phone model?

(b) Name the user who has made maximum call to a user (say Y, here Y can be called by many user).