# EXPERIMENT NUMBER –Practical 6.1

Name- Anushka Rai

UID-20BCS4622

Class- 20IOT1

Section - B

TOPIC OF EXPERIMENT –LEARN HOW TO USE CLASSES USING C+

AIM OF EXPERIMENT: Write a program that takes information about institute staff information for

FLOWCHART/ ALGORITHM

1) Teacher code, name, subject and publication

2) Officer code, name and grade

3) Typist code, name, speed and daily wages and

displays it using hierarchal inheritance

ALGORITHM:

Step 1: Start

Step 2: Create a class staff which enters and prints the name and code.

Step 3: Create a class teacher to which takes input for subject and publication and displays the same and also, inherit class staff.

Step 4: Similarly, create a class officer to print grade and inherit class staff.

Step 5: Create class typist to print speed and inherit class staff.

Step 6: Create another class casual to print wages and inherit class staff.

Step 7: Inside the main method, create objects for classes teacher, officer and casual.

Step 8: Create a while loop to display the main menu repeatedly.

Step 9: Create a menu driven approach for each, teacher, officer and typist.

Step 10: For each of them, create another menu driven approach to select among create, display or back to main menu.

Step 11: Inside the create option, ask the user if they are interested in entering the data and proceed further accordingly.

Step 12: In the display section, print the information entered in create menu.

Step 13: In the third option, send the user back to main menu using appropriate label.

Step 14: In the default section, print invalid choice.

Step 15: Stop

PROGRAM CODE

#include <iostream>

#include <conio.h>

using namespace std;

class staff

{

protected:

int code;

char name[20];

public:

void getstaff(void)

{

cout<<"\n\nEnter code :-";

cin>>code;

cout<<"Enter name :-";

cin>>name;

}

void dispstaff(void)

{

cout<<"\nNAME :-"<<name;

cout<<"\nCODE :-"<<code;

}

};

class teacher : public staff

{

char sub[20];

char pub[20];

public:

void create(void)

{

getstaff();

cout<<"Enter Subject :-";

cin>>sub;

cout<<"Enter Publication :-";

cin>>pub;

}

void display(void)

{

dispstaff();

cout<<"\nSUBJECT :-"<<sub;

cout<<"\nPUBLICATION:-"<<pub;

}

};

class officer : public staff

{

char grade;

public:

void create(void)

{

getstaff();

cout<<"Enter Grade :-";

cin>>grade;

}

void display(void)

{

dispstaff();

cout<<"\nGRADE :-"<<grade;

}

};

class typist : public staff

{

float speed;

public:

void gettypist(void)

{

getstaff();

cout<<"Enter speed (wpm):-";

cin>>speed;

}

void disptypist(void)

{

dispstaff();

cout<<"\nSPEED :-"<<speed;

}

};

class casual : public typist

{

float dailywages;

public:

void create(void)

{

gettypist();

cout<<"Enter Daily Wages :-";

cin>>dailywages;

}

void display(void)

{

disptypist();

cout<<"\nDAILY WAGES:-"<<dailywages;

}

};

int main()

{

teacher o1t[10];

casual o1c[10];

officer o1o[10];

int choice,i;

char test;

while(1)

{

int count;

start:

cout<<"\n=====EDUCATION INSTITUTION DATABASE=====\n\n\n";

cout<<"Choose Category of Information\n";

cout<<"1) Teachers\n";

cout<<"2) Officer\n";

cout<<"3) Typist\n";

cout<<"4) Exit\n";

cout<<"Enter your choice:-";

cin>>choice;

switch(choice)

{

case 1 : while(1)

{

cout<<"\n=====TEACHERS INFORMATION=====\n\n";

cout<<"\nChoose your choice\n";

cout<<"1) Create\n";

cout<<"2) Display\n";

cout<<"3) Jump to Main Menu\n";

cout<<"Enter your choice:-";

cin>>choice;

switch(choice)

{

case 1 : for(count=0,i=0;i<10;i++)

{

cout<<endl;

o1t[i].create();

count++;

cout<<endl;

cout<<"\n\nAre you Interested in entering data\n";

cout<<"Enter y or n:-";

cin>>test;

if(test=='y' || test=='Y')

continue;

else goto out1;

}

out1:

break;

case 2 : for(i=0;i<count;i++)

{

cout<<endl;

o1t[i].display();

cout<<endl;

}

getch();

break;

case 3 : goto start;

default: cout<<"\nEnter choice is invalid\ntry again\n\n";

}

}

case 2 : while(1)

{

cout<<"\n=====OFFICERS INFORMATION=====\n\n";

cout<<"\nChoose your choice\n";

cout<<"1) Create\n";

cout<<"2) Display\n";

cout<<"3) Jump to Main Menu\n";

cout<<"Enter your choice:-";

cin>>choice;

switch(choice)

{

case 1 : for(count=0,i=0;i<10;i++)

{

cout<<endl;

o1o[i].create();

count++;

cout<<endl;

cout<<"\n\nAre you Interested in entering data\n";

cout<<"Enter y or n:-";

cin>>test;

if(test=='y' || test=='Y')

continue;

else goto out2;

}

out2:

break;

case 2 : for(i=0;i<count;i++)

{

cout<<endl;

o1o[i].display();

cout<<endl;

}

getch();

break;

case 3 : goto start;

default: cout<<"\nInvalid choice\ntry again\n\n";

}

}

case 3 : while(1)

{

cout<<"\n=====TYPIST INFORMATION=====\n\n";

cout<<"\nChoose your choice\n";

cout<<"1) Create\n";

cout<<"2) Display\n";

cout<<"3) Jump to Main Menu\n";

cout<<"Enter your choice:-";

cin>>choice;

switch(choice)

{

case 1 : for(count=0,i=0;i<10;i++)

{

cout<<endl;

o1c[i].create();

count++;

cout<<endl;

cout<<"\n\nAre you Interested in entering data\n";

cout<<"Enter y or n:-";

cin>>test;

if(test=='y' || test=='Y')

continue;

else goto out3;

}

out3:

break;

case 2 : for(i=0;i<count;i++)

{

cout<<endl;

o1c[i].display();

cout<<endl;

}

getch();

break;

case 3 : goto start;

default: cout<<"\nInvalid choice\ntry again\n\n";

}

}

case 4 : goto end;

}

}

end:

return 0;

}

TROUBLESHOOTING: No errors

PROGRAMS’ EXPLANATION - In this program, we are required to display staff information about teachers, officers and typists using multiple inheritance. For this purpose, I have created a class named staff which takes user input for common details among all the staff such as name and code. Then I inherited this class in all the other classes. I created separate classes for taking rest of the information of teachers, officers and typists individually. Finally in the main method, I created a menu driven approach which enables user to enter whose information they would like to display among teachers, officers and typists. In the individual sections, I again created a menu driven approach to select among create display and return to main menu which helps user to enter the information, display the entered information or return to main menu respectively. This program could be successfully implemented using multiple inheritance and saved time and memory

OUTPUT

# EXPERIMENT NUMBER –Practical 6.2

STUDENT’S NAME -Anushka Rai

STUDENT’S UID – 20BCS4622

CLASS AND GROUP – CSEIOT B

SEMESTER – EVEN

AIM OF THE EXPERIMENT Create a class student having student uid and getnumber(),putnumber() as member functions to get the values and display it. Derive a class test having marks in different subjects and getmarks() and putmarks() as member functions to get and display the values. Derive another class sports from student class having sports score and getscore(), putscore() as member functions to get and display the values. Derive a class result from test

and sports class and define a function display() to calculate total marks. Implement it with the object of result class. If it gives any error, resolve it by adding the required functionality.

FLOWCHART/ ALGORITHM

ALGORITHM for the above task:

1. Start
2. Create a class student to intake and print roll number.
3. Create a derived class test with class student as virtual base class to take input of marks and print them.
4. Similarly, create a derived class sport with student as virtual base class to intake scores and print them.
5. Now create a derived class result with sport and test as base classes to calculate the total score and print it along with total marks and total scores of sports.
6. In the main section, create the object for class result and call the funtions to display total marks, total sport score as well as result.
7. Stop.

PROGRAM CODE

**#include<iostream>**

**#include<conio.h>**

**using namespace std;**

**class student {**

**int rno;**

**public:**

**void getnumber() {**

**cout << "Enter Roll No:";**

**cin>>rno;**

**}**

**void putnumber() {**

**cout << "\n\n\tRoll No:" << rno << "\n";**

**}**

**};**

**class test : virtual public student {**

**public:**

**int part1, part2;**

**void getmarks() {**

**cout << "Enter Marks\n";**

**cout << "Part1:";**

**cin>>part1;**

**cout << "Part2:";**

**cin>>part2;**

**}**

**void putmarks() {**

**cout << "\tMarks Obtained\n";**

**cout << "\n\tPart1:" << part1;**

**cout << "\n\tPart2:" << part2;**

**}**

**};**

**class sports : public virtual student {**

**public:**

**int score;**

**void getscore() {**

**cout << "Enter Sports Score:";**

**cin>>score;**

**}**

**void putscore() {**

**cout << "\n\tSports Score is:" << score;**

**}**

**};**

**class result : public test, public sports {**

**int total;**

**public:**

**void display() {**

**total = part1 + part2 + score;**

**putnumber();**

**putmarks();**

**putscore();**

**cout << "\n\tTotal Score:" << total;**

**}**

**};**

**int main() {**

**result obj;**

**obj.getnumber();**

**obj.getmarks();**

**obj.getscore();**

**obj.display();**

**return 0;**

**}**

ERRORS ENCOUNTERED DURING PROGRAM’S EXECUTION

(Kindly jot down the compile time errors encountered)

N/A

PROGRAMS’ EXPLANATION (in brief)

The program is performed by creating a class to intake the roll number of the student and print it. Then two virtual derived class with base class student are created as test and sport to intake marks and score of the student and display them respectively. Then a derived class result is created with sport and test as base classes to calculate total score. In the main method the object for result class is created and all the information is printed.

OUTPUT

