v)Hybrid inheritance

#include<iostream>

using namespace std;

class student

{

protected:

string name;

string branch;

void getst()

{

cout<<"enter name and branch";

cin>>name>>branch;

}

void showst()

{

cout<<"name="<<name<<endl;

cout<<"branch="<<branch<<endl;

}

};

class marks:public student

{

protected:

int m1,m2,m3,m4;

int pinno;

void getm()

{

getst();

cout<<"enter the student marks";

cin>>m1>>m2>>m3>>m4;

cout<<"enter pinno";

cin>>pinno;

}

void showm()

{

showst();

cout<<"pinno="<<pinno<<endl;

}

};

class project

{

protected:

int ip,ep;

void getpro()

{

cout<<"enter project(internal) marks";

cin>>ip;

cout<<"enter project(external)marks";

cin>>ep;

}

};

class percentage:public marks,public project

{

private:

float per;

public:

void result()

{

getm();

getpro();

per=(float)(m1+m2+m3+m4+ip+ep)/6;

showm();

cout<<"percentage="<<per<<endl;

}

};

int main()

{

percentage p;

p.result();

return 0;

}

Output:

