**EVEREST ENGINEERING COLLEGE**

**SANEPA, LALITPUR**



(AFFILIATED TO POKHARA UNIVERSITY)

AN

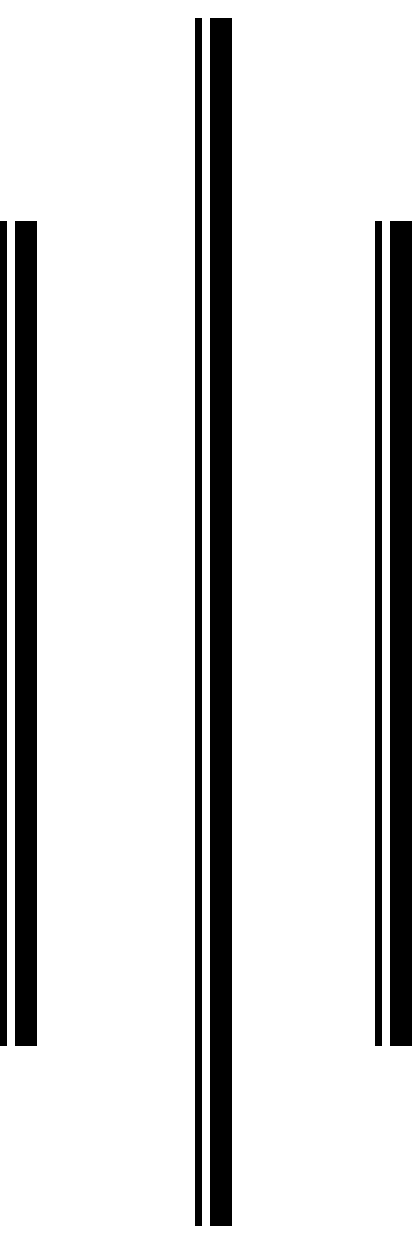
LAB REPORT

ON

**Object Oriented Programming In C++**

***[Constructor In Inheritance]***

**Lab Sheet: 8**



**SUBMITTED BY SUBMITTED TO**

Name:-**SAKCHYAM ACHARYA** Er.Pradip Paudel

Roll No:-40 Everest Engineering College

Department of Computer Engineering

Faculty:-BE CMP

Year/Semester: - 2nd SEM

Batch: 2021

Verified By:-

Student's Signature

**Index Page:-**

Topics Page No

* Title, Objective....................................................................................2
* Theory……………………………………………………………………………….2

1) Constructor & Destructor In Inheritance

2) Argument passing mechanism for supplying initial values

to the base class constructor

* Questions with Source Code & Output…………………………………………...2-10
* Discussion & Conclusion ...................................................................................10

**Title:** **Constructor In Inheritance.**

**Objective:**

• To be familiar with the use of constructor in inheritance.

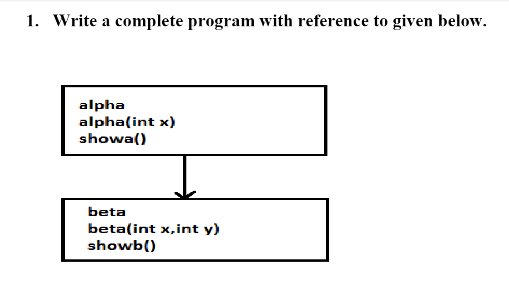
• To understand the working mechanism of destructor & constructor in inheritance.

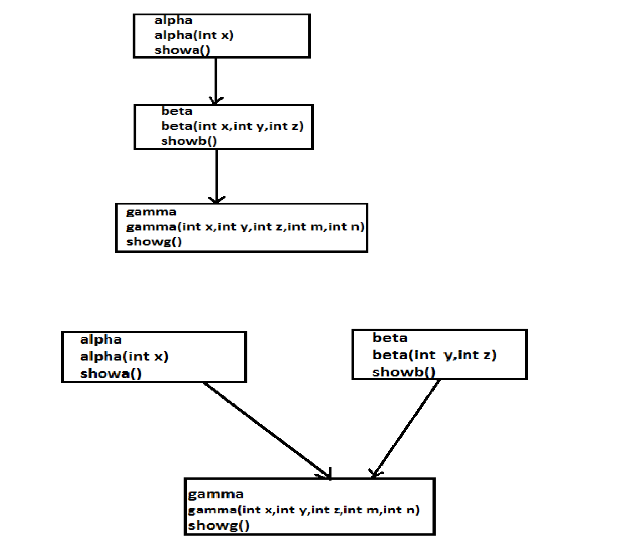
* **Theory:**

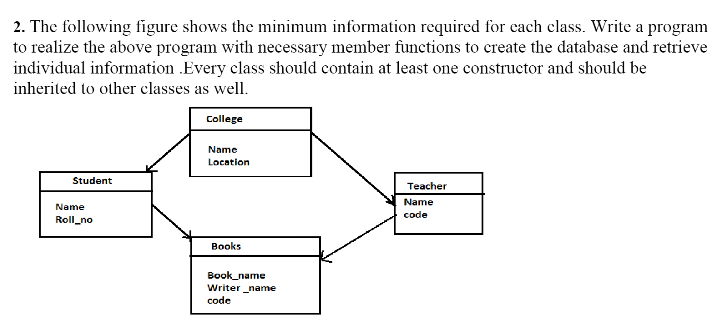
• **Constructor & Destructor In Inheritance:-** Constructor and Destructor are the special member functions of the class which are created by the C++ compiler or can be defined by the user. **Constructor is used to initialize the object of the class while destructor is called by the compiler when the object is destroyed**.

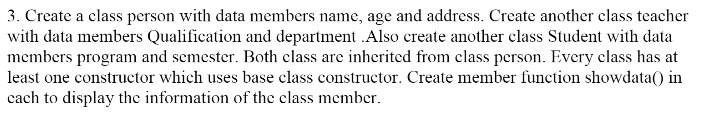
• **Argument passing mechanism for supplying initial values to the base class constructor:-**

* **Problems:-**

****

****





**//Problem No(1.a)**

**//SOURCE CODE**

#include<iostream>

using namespace std;

class alpha{

protected:

int a;

public:

alpha(int x){

a=x;

}

void showa(){

cout<<"Value of a="<<a;

}

};

class beta:public alpha{

protected:

int b;

public:

beta(int x,int y):alpha(x){

b=y;

}

void showb(){

cout<<"\nValue of b="<<b;

}

};

int main(){

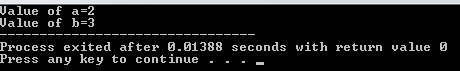
beta obj(2,3);

obj.showa();

obj.showb();

return 0;

}



**//Problem No(1.b)**

**//SOURCE CODE**

#include<iostream>

using namespace std;

class alpha{

protected:

int a;

public:

alpha(int x){

a=x;

}

void showa(){

cout<<"Value of a="<<a;

}

};

class beta:public alpha{

protected:

int b,c;

public:

beta(int x,int y,int z):alpha(x){

b=y;

c=z;

}

void showb(){

cout<<"\nValue of b="<<b;

cout<<"\nValue of c="<<c;

}

};

class gamma:public beta{

protected:

int d,e;

public:

gamma(int x,int y,int z,int m,int n):beta(x,y,z){

d=m;

e=n;

}

void showg(){

cout<<"\nValue of d="<<d;

cout<<"\nValue of e="<<e;

}

};

int main(){

gamma obj(1,2,3,4,5);

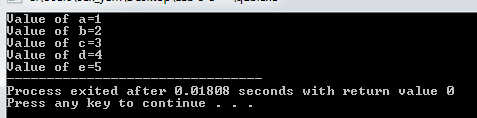
obj.showa();

obj.showb();

obj.showg();

return 0;

}

****

**//Problem No(1.c)**

**//SOURCE CODE**

#include<iostream>

using namespace std;

class alpha{

protected:

int a;

public:

alpha(int x){

a=x;

}

void showa(){

cout<<"Value of a="<<a;

}

};

class beta{

protected:

int b,c;

public:

beta(int y,int z){

b=y;

c=z;

}

void showb(){

cout<<"\nValue of b="<<b;

cout<<"\nValue of c="<<c;

}

};

class gamma:public alpha,public beta{

protected:

int d,e;

public:

gamma(int x,int y,int z,int m,int n):alpha(x),beta(y,z){

d=m;

e=n;

}

void showg(){

cout<<"\nValue of d="<<d;

cout<<"\nValue of e="<<e;

}

};

int main(){

gamma obj(1,2,3,4,5);

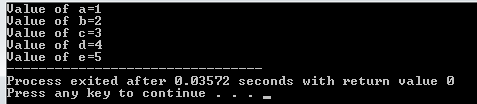
obj.showa();

obj.showb();

obj.showg();

return 0;

}



**//Problem No(2)**

**//SOURCE CODE**

#include<iostream>

#include<string.h>

using namespace std;

class college{

protected:

char name[40],location[20];

public:

college(char na[],char loca[]){

strcpy(name,na);

strcpy(location,loca);

}

void display(){

cout<<"College Name="<<name<<"\nLocation="<<location;

}

};

class student:virtual public college{

protected:

char name[25];

short int roll\_no;

public:

student(char na[],char loca[],char nam[],short int ro):college(na,loca){

strcpy(student::name,nam);

roll\_no=ro;

}

void display(){

cout<<"\n\nStudent's Name="<<name<<endl<<"Roll No="<<roll\_no;

}

};

class teacher:virtual public college{

protected:

char name[25];short int code;

public:

teacher(char na[],char loca[],char namee[],short int co):college(na,loca){

strcpy(teacher::name,namee);

code=co;

}

void display(){

cout<<"\n\nTeacher's Name="<<name<<endl<<"Code="<<code;

}

};

class books:public student,public teacher{

protected:

char book\_name[25],writer\_name[20];

char code[20];

public:

books(char na[],char loca[],char nam[],short int ro,char namee[],short int co,char bn[],char wn[],char cod[]):student(na, loca,nam,ro) , teacher(na,loca,namee,co),college(na,loca){

strcpy(book\_name,bn);

strcpy(writer\_name,wn);

strcpy(books::code,cod);

}

void display(){

college::display();

student::display();

teacher::display();

cout<<"\n\nBook Name="<<book\_name<<endl<<"Writer Name="<<writer\_name<<endl<<"Code="<<code;

}

};

int main(){

char cn[]="Everest Engineering College" ;char cl[]="Sanepa,Lalitpur" ; char sn[]="Sujit Tamang" ; char tn[]="Pradip Paudel" ; char bn[]="OOP IN CPP" ; char wn[]="Denis Richhi" ; char tc[]="23-18222-7171";

short int sr=16;

short int co=721;

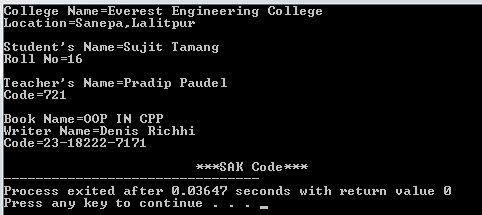
books obj(cn,cl,sn,sr,tn,co,bn,wn,tc);

obj.display();

cout<<"\n\n\t\t\t\*\*\*SAK Code\*\*\*";

return 0;

}



**//Problem No(3)**

**//SOURCE CODE**

#include<iostream>

#include<string.h>

using namespace std;

class student{

protected:

char program[15],semester[5];

public:

student(char pro[],char sem[]){

strcpy(program,pro);

strcpy(semester,sem);

}

void showdata(){

cout<<"\nProgram="<<program<<"\nSemester="<<semester;

}

};

class teacher{

protected:

char qualification[15],department[15];

public:

teacher(char quali[],char depart[]){

strcpy(qualification,quali);

strcpy(department,depart);

}

void showdata(){

cout<<"\n\nQualification="<<qualification<<"\nDepartment="<<department;

}

};

class person:public teacher,public student{

protected:

char name[20],address[20];

short int age;

public:

person(char pro[],char sem[],char quali[],char depart[],char na[],char add[],short int ag):student(pro,sem),teacher(quali,depart){

strcpy(name,na);

strcpy(address,add);

age=ag;

}

void showdata(){

cout<<"\n\t\t\tStudent's Details";

student::showdata();

cout<<"\n\t\t\tTeacher's Details";

teacher::showdata();

cout<<"\n\n\t\t\tPerson's Details";

cout<<"\n\nName="<<name<<"\nAddress="<<address<<"\nAge="<<age;

}

};

int main(){

char pro[]="Computer" , sem[]="2nd" ,qua[]="Post Graduate" ,depa[]="Civil,Computer" ,na[]="SAK Wheels",add[]="Sanagaun,Lalitpur";

short int ag=20;

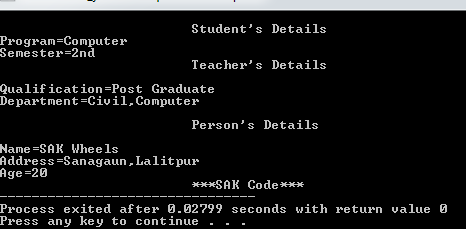
person obj(pro,sem,qua,depa,na,add,ag);

obj.showdata();

cout<<"\n\t\t\t\*\*\*SAK Code\*\*\*";

return 0;

}



**Discussion & Conclusion:-**

The program is focused on various tasks on “**Constructor In Inheritance**”. From this program I understood how to use constructor in inheritance & it’s working mechanism too.

**\* Thank You\*[SAK Wheels]**