МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра ЕОМ



Звіт до лабораторних робіт

з ООП

Виконав:

ст. групи КІ-15

Ковальський П.А.

Прийняв:

викл. Козак Н.Б.

Львів-2020

ЛАБОРАТОРНА РОБОТА № 6.

СПАДКУВАННЯ

**Мета:** познайомитися із спадкуванням класів.

**ТЕОРЕТИЧНІ ВІДОМОСТІ**

**Спадкування (ієрархія "іs a")**

Спадкування – це механізм, за допомогою якого один клас може одержувати атрибути та функціональність іншого. Спадкування дозволяє створювати ієрархію класів. При створенні нового класу, що повністю дублює існуючий клас і дещо розширяє його новими властивостями і функціональністю програміст може не дублювати існуючий клас і дописувати в нього розширену функціональність, а вказати, що новий клас є спадкоємцем елементів попередньо визначеного класу і визначити у ньому лише необхідну нову функціональність. В цьому випадку існуючий клас, функціональність якого розширюється у новому класі, називається базовим класом (base class). Новостворений клас називається похідним класом (derived class), або спадкоємцем. Кожен похідний клас може бути використаним у ролі базового класу для майбутніх похідних класів створюючи при цьому дерево спадкування, яке ще називають ієрархією спадкування класів (class hierarchy). Спадкування прийнято відображати у вигляді графу (дерева) у напрямку зверху-вниз. При цьому клас, що є у самому верху є самим першим базовим класом і називається кореневим класом або коренем дерева спадкування класів. Похідний клас, через проміжний, може наслідувати характеристики базового класу. У цьому випадку говорять, що базовий клас є непрямим базовим класом (indirect base class) для похідного. Зокрема, корінь дерева наслідувань є непрямим базовим класом для усіх класів, які знаходяться нижче першого рівня ієрархії. Клас, який При одиночному спадкуванні (single inheritance) клас породжується одним базовим класом. При множинному спадкуванні (multiple inheritance) похідний клас успадковує властивості декількох базових класів, причому можлива ситуація коли один базовий клас буде успадкований кілька разів по кількох гілках. При створенні об'єкта похідного класу в пам'яті зберігаються копії усіх класів, які становлять вітку, що породила даний клас.

97

Похідний клас наслідує характеристики базового через специфікатор доступу (acces specifier) : "двокрапка". Нижче наведено синтаксис спадкування базового класу:

Конструктори при одинарному спадкуванні

При створенні об'єкту класу автоматично викликається його конструктор. Якщо об'єкт породжується від похідного класу, то, очевидно, при створенні об‘єкту похідного класу повинен викликатись і конструктор базового класу. У цьому випадку порядок виклику і виконання конструкторів є таким:

1. при створенні об'єкта похідного класу генерується виклик конструктора цього класу, але сам конструктор не виконується;

2. після генерації виклику конструктора похідного класу генерується виклик і виконується конструктор базового класу;

3. по закінченню виконання тіла конструктора базового класу виконується тіло конструктора похідного класу.

**Варіант-12**



**Код програми:**

**main.cpp:**

#include <iostream>

#include "water.h"

#include "sea.h"

using namespace std; int main() {

Water MyWater;

Sea MySea;

string country, name; double ppm;

int choice,mind, maxd, s, v;

while(true) {

cout<<"Ввести Дані про Водойму – Натисніть 1\n";

cout<<"Ввести Дані про Море - Натисніть 2\n";

cout<<"Показати Дані - Натисніть 3\n";

cout<<"Вихід – Натисніть Будь Яку Іншу Клавішу\n";

cin >> choice;

if(choice == 1) {

cout<<"Введіть Мінімальну глибину\n";

cin>>mind;

cout<<"Введіть Мксимальну глибину\n";

cin>>maxd;

cout<<"Введіть Площу Водойми\n";

cin>>s;

cout<<"Введіть Об'єм Водойми\n";

cin>>v;

MyWater.SetMinDepth(mind);

MyWater.SetMaxDepth(maxd);

MyWater.SetSquare(s);

MyWater.SetVolume(v); }

if(choice == 2){

cout<<"Введіть Мінімальну глибину\n";

cin>>mind;

cout<<"Введіть Мксимальну глибину\n";

cin>>maxd;

cout<<"Введіть Площу Водойми\n";

cin>>s;

cout<<"Введіть Об'єм Водойми\n";

cin>>v;

cout<<"Введіть Назву Моря\n";

cin>>name;

cout<<"Введіть Назву Країни\n";

cin>>country;

cout<<"Введіть Солоність Моря\n";

cin>>ppm;

MySea.SetMinDepth(mind);

MySea.SetMaxDepth(maxd);

MySea.SetSquare(s);

MySea.SetVolume(v);

MySea.SetName(name);

MySea.SetCountry(country);

MySea.SetSaltPpm(ppm);

MySea.Verify(); }

if (choice==3){

cout<<"Водойма:\n";

MyWater.Print();

cout<<"----------------------\n";

cout<<"Море:\n";

MySea.Print();

MySea.Show(); }

if (choice>3 || choice<1) break; }

return 0; }

**water.h:**

#ifndef GRANDPARENT\_H

#define GRANDPARENT\_H

#include <iostream>

#include <string>

using namespace std;

class Water {

protected:

int minDepth;

int maxDepth;

int square;

int volume;

public:

Water(){

minDepth = 2;

maxDepth = 20;

square = 1000;

volume = 50000;

}

~Water(){}

void SetMinDepth(int c)

{

minDepth = c;

if (minDepth<0)

minDepth = 0;

}

int GetMinDepth()

{

return minDepth;

}

void SetMaxDepth(int c)

{

maxDepth = c;

if (maxDepth<0)

maxDepth = 0;

}

int GetMaxDepth()

{

return maxDepth;

}

void SetSquare(int c)

{

square = c;

if (square<0)

square = 0;

}

int GetSquare(){

return square;

}

void SetVolume(int c)

{

volume = c;

if (volume<0)

volume = 0;

}

int GetVolume(){

return volume;

}

void Print()

{

cout<<"Мінімальна Глибина: "<<GetMinDepth()<<"м."<<endl;

cout<<"Максимальна Глибина: "<<GetMaxDepth()<<"м."<<endl;

cout<<"Площа: "<<GetSquare()<<"м^2"<<endl;

cout<<"Об'єм: "<<GetVolume()<<"м^3"<<endl;

}

};

#endif

**sea.h:**

#include <iostream>

#include <string>

#include "water.h"

using namespace std;

class Sea :

public Water

{

private:

double saltPpm;

string name;

string country;

public: Sea()

{

saltPpm = 42;

name = "BlackSea";

country = "Ukraine";

}

~Sea(){}

void SetSaltPpm(double ppm)

{

saltPpm = ppm;

}

double GetSaltPpm()

{

return saltPpm;

}

void SetName(string n)

{

name = name;

}

string GetName()

{

return name;

}

void SetCountry(string c)

{ country = c;

}

string GetCountry()

{

return country;

}

void Verify()

{

if(saltPpm<0) saltPpm = 0;

if(saltPpm>1000) saltPpm = 1000;

if(name=="") name = "BlackSea";

if(country=="") country= "Ukraine";

}

void Show(){

cout<<"Назва Моря: "<<GetName()<<endl;

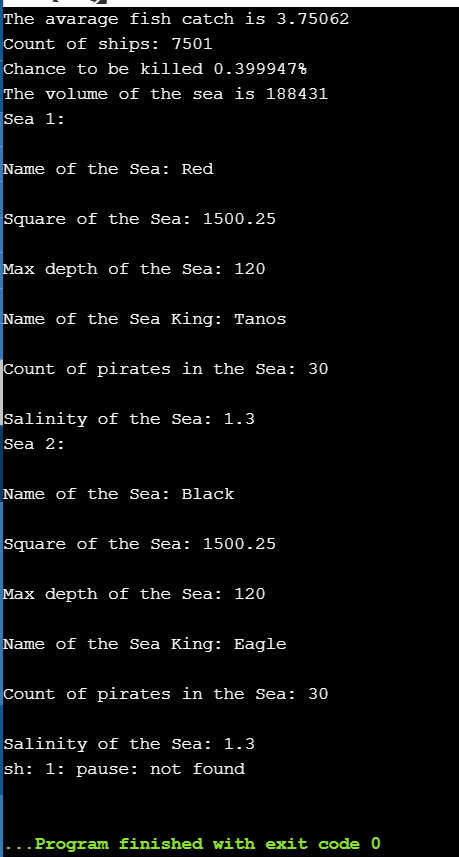
cout<<"Країна "<<GetCountry()<<endl;

cout<<"Солоність = "<<GetSaltPpm()<<" Проміле\n";

}

};

**Результат**



**Висновок:** Роблячи цю лабораторну роботу я познайомився із спадкуванням.