

Лабораторна робота №5.**Тема: Успадкування. Віртуальні функції. Ієрархія й контейнерні класи.****Методичні вказівки****1. Успадкування**

Одним найважливішим досягненням мови **C++** є можливість будувати ієрархічні структури, що розвиваються, програмних комплексів. Успадкування надає можливість породжувати нові класи на базі вже існуючих з передачею породженням класам спадщини у вигляді даних і членів-функцій батьківського класу (або класів, якщо прямих декілька батьків). Породжені класи мають можливість розширювати набір даних, отриманих у спадщину, модифікувати батьківські методи й функції, створювати нові дані та нові функції їх обробки.

Клас, який успадковується, називається базовим(предком). Клас, який успадковує базовий клас, називається похідним(дочірнім). Механізм успадкування (**inheritance**) передбачає дві можливості. У першому випадку, який називають простим успадкуванням, батьківський клас один. У другому випадку батьків два або більше процес, що й відповідає, іменують терміном множинне спадкування. Клас може успадковуватися довільною кількістю інших класів.

Синтаксис успадкування:

```
[атрибути] class|struct|union ім'я_класу
[: [virtual] public|private|protected ім'я_базового_класу1,
[virtual] public|private|protected ім'я_базового_класу2, ...] {
тіло похідного класу
} [об'єкти класу];
```

Конструктори і деструктори

При створенні об'єктів похідного класу необхідно спочатку створити проміжний об'єкт базового класу. Це зрозуміло — об'єкт похідного класу є модифікований об'єкт базового класу. Він не може виникнути з повітря. Отже, спочатку викликається конструктор базового класу, а потім — конструктор похідного класу. Якщо базових класів декілька, вони викликаються в порядку їх перерахування в списку успадкування класів. Деструктори викликаються в зворотному порядку.

Конструктора похідного класу має синтаксис :

```
ім'я_похідного_класу : ім'я_базового_класу1(параметри) , ...,
ім'я_базового_класу(параметри)
{
тіло конструктора
}
```

Конструктори без параметрів базових класів можуть викликатися неявно.

Приклад виклику конструкторів та деструкторів.

```
#include "stdafx.h"
#include<iostream>
using namespace std;
class Inner
{
public:
    long inf_inner;
    Inner() : inf_inner(0) {
        cout << "Default constructor Inner\n";
    }
    Inner(long n) : inf_inner(n) {
        cout << "Constructor with (long n) Inner\n";
    }
    Inner(Inner& s) { inf_inner = s.inf_inner; cout<<"Constructor Copy Inner\n";
    }
```

```

    ~Inner() { cout << "Destructor Inner\n"; }
};
class Base1
{
public:
    double inf_base1;
    Inner inf_in_inner;
    Base1() : inf_in_inner(0), inf_base1(0) {
        cout << "Default constructor Base1\n";
    };
    Base1(long x, double y) : inf_in_inner(x), inf_base1(y) {
        cout << "Constructor with (long x, double y) Base\n";
    }
    Base1(Base1& s) : inf_in_inner(s.inf_in_inner), inf_base1(s.inf_base1) {
        cout << "Constructor Copy Base1"; }
    ~Base1() { printf("Destructor Base1\n"); }
    void printBase() { cout<<"Base1::Inner::inf_inner ="<< inf_in_inner.inf_inner
<< " inf_base1 ="<<inf_base1<<endl; }
};
class Base2
{
public:
    int inf_base2;
    Base2() : inf_base2(0) {
        cout << "Default constructor Base2\n";
    };
    Base2(int i) : inf_base2(i) {
        cout << "Constructor with (int ) Base\n";
    }
    Base2(Base2& s) : inf_base2(s.inf_base2) {
        cout << "Constructor Copy Base2 \n";
    }
    ~Base2() { cout<<"Destructor Base2\n"; }
};
class Derived : public Base1, public Base2
{
public:
    float inf_drv;
    Derived() : inf_drv(0) { cout << "Default constructor Derived \n"; }
    Derived(long inner, double base1, int base2, float drv)
    : Base1(inner,base1), Base2(base2), inf_drv(drv) {
        cout << "Constructor with (... ) Derived \n";
    }
    ~Derived()
    {
        cout << "Destructor Derived \n";
    }
};
int main()
{
    Derived a;
    Derived b(1,2.1,3,4.1);
    return 0;
}

```

2. Віртуальні функції.

В основі механізму динамічного поліморфізму лежить механізм успадкування і віртуальних функцій. Віртуальна функція з'являється в базовому класі за допомогою ключового слова `virtual`. У похідних класах слово `virtual` можна не вживати. Віртуальні функції можна перевизначити в кожному з похідних класів, налагодивши її на вирішення нової задачі. Якщо

функція викликається з об'єкта відповідного класу, що входить в ієрархію, її поведінка нічим не відрізняється від звичайної функції-члена. Віртуальність має зміст лише при виклику через вказівник на базовий клас викликається функція з похідного класу. На віртуальні функції накладаються наступні обмеження:

- 1) не можуть бути статичними;
- 2) не можуть бути дружніми;
- 3) конструктори не можуть бути віртуальними.

Оскільки об'єкти похідного класу, за суттю, є модифікованими об'єктами базового класу, на них у мові C++ можна посилатися за допомогою вказівника на об'єкти базового класу. Якщо вони містять віртуальні функції, їх вибір буде виконаний на етапі виконання програми. Ієрархія класів, зв'язаних визначеною віртуальною функцією, називається поліморфичним кластером.

Приклад.

Завдання до лабораторної роботи:

Розробити та реалізувати класи згідно варіантів. Передбачити введення початкових даних: з клавіатури, файлу та використовуючи датчик випадкових чисел. Написати тестові програми для кожної задачі.

Завдання 1. Варіанти задач. *Ієрархія.*

Задача 1.1. Створити клас студент, що має ім'я (вказівник на рядок), курс і ідентифікаційний номер. Визначити конструктори, деструктор і функцію друку. Створити public-похідний клас - студент-дипломник, що має тему диплома. Визначити конструктори за замовчуванням і з різним числом параметрів, деструктори, функцію друку. Визначити функції перепризначення назви диплома й ідентифікаційного номера.

Задача 1.2. Створити клас тварина, що має класифікацію (вказівник на рядок), число кінцівок, число нащадків. Визначити конструктори, деструктор і функцію друку. Створити public-похідний клас - свійська тварина, що має кличку. Визначити конструктори за замовчуванням і з різним числом параметрів, деструктори, функцію друку. Визначити функції перепризначення назви клички й числа нащадків.

Задача 1.3. Створити клас машина, що має марку (вказівник на рядок), число циліндрів, потужність. Визначити конструктори, деструктор і функцію друку. Створити public-похідний клас - вантажівка, що має вантажопідйомність кузова. Визначити конструктори за замовчуванням і з різним числом параметрів, деструктори, функцію друку. Визначити функції перепризначення марки й вантажопідйомності.

Задача 1.4. Створити клас карта, що має ранг і масть. Карту можна перевернути й відкрити. Створити клас - колода карт, що містить карти. Створити два похідних класу від колоди карт, в одному карти можуть діставатися тільки один по одному, в іншому - витягатися довільно.

Задача 1.5. Створіть клас точка, яка має координати. Клас еліпсів і клас кіл. Визначити ієрархію типів. Визначити функції друку, конструктори, деструктори, обчислення площі.

Задача 1.6. Створити клас чотирикутників, квадратів і прямокутників. Створити з них ієрархію. Визначити функції друку, конструктори й деструктори, обчислення площі й периметра.

- Задача 1.7. Створити клас транспортні засоби, автомобіль, вантажівка, пароплав і літак. Створити з них ієрархію. Визначити функції друку, конструктори й деструктори.
- Задача 1.8. Створити клас рідина, що має назва (вказівник на рядок), щільність. Визначити конструктори, деструктор і функцію друку. Створити public-похідний клас - спиртні напої, що має міцність. Визначити конструктори за замовчуванням і з різним числом параметрів, деструктори, функцію друку. Визначити функції перепризначення щільності й міцності.
- Задача 1.9. Створити клас людей, що має ім'я (вказівник на рядок), вік, вага. Визначити конструктори, деструктор і функцію друку. Створити public-похідний клас - школяр, що має клас (рік навчання). Визначити конструктори за замовчуванням і з різним числом параметрів, деструктори, функцію друку. Визначити функції перепризначення віку й класу.
- Задача 1.10. Створити клас вікно, що має координати верхнього лівого й нижнього правого кута, колір тла (вказівник на рядок). Визначити конструктори, деструктор і функцію друку. Створити public-похідний клас - вікно з меню, що має рядок. Визначити конструктори за замовчуванням і з різним числом параметрів, деструктори, функцію друку. Визначити функції перепризначення кольору тла й рядка меню.
- Задача 1.11. Створити клас точка, що має координати. Визначити конструктори, деструктор і функцію друку. Створити public-похідний клас - кольорова точка колір, що має, крапки . Визначити конструктори за замовчуванням і з різним числом параметрів, деструктори, функцію друку. Визначити функції перепризначення кольору й координат крапки, виводу крапки на екран.
- Задача 1.12. Створити клас людей, що має ім'я (вказівник на рядок), вік, вага. Визначити конструктори, деструктор і функцію друку. Створити public-похідний клас - повнолітні номер, що має, паспорта. Визначити конструктори за замовчуванням і з різним числом параметрів, деструктори, функцію друку. Визначити функції перепризначення віку й номера паспорта.
- Задача 1.13. Створити колоду карт. Конструктори колоди повинні ініціалізувати колоду впорядковане й випадковим образом. Створити похідний клас від колоди - пасьянс, у яким вибираються по три карти й, якщо дві крайні одного кольору, то їх викидають. Усю колоду проходять три рази.
- Задача 1.14. Створити клас точка й похідні від нього - коло і еліпс. Визначити конструктори, деструктор і функцію друку. Визначити функції пере-установки центру окружності й еліпса.

Завдання 2. Варіанти задач. **Композиція та ієрархія.**

- Задача 2.1. Створити клас кімната, що має площа. Визначити конструктор і метод доступу. Створити клас однокімнатних квартира, що містить кімнату й кухню (її площа), поверх (кімната втримується в класі однокімнатна квартира). Визначити конструктори, методи доступу. Визначити public-похідний клас однокімнатних квартир різних міст (додатковий параметр - назва міста). Визначити конструктори, деструктор і функцію друку.
- Задача 2.2. Створити клас хвіст, що має довжину. Визначити конструктори й метод доступу. Створити клас собака, що містить клас хвіст. Додатково є колір (вказівник на рядок), вік. Визначити конструктори й деструктор. Визначити public- похідний клас собака, що має додатково кличку(вказівник на рядок). Визначити конструктори, деструктори й функцію друку.
- Задача 2.3. Створити клас двигун, що має потужність. Визначити конструктори й метод доступу Створити клас машин, що містить клас двигун. Додатково є марка

(вказівник на рядок), ціна. Визначити конструктори й деструктор. Визначити public-похідний клас вантажівка, що має додатково вантажопідйомність. Визначити конструктори, деструктори й функцію друку.

Задача 2.4. Використовуючи ієрархію й композицію класів, створити бінарне дерево. У бінарного дерева є кореневий вузол. Ми можемо вставляти вузол. Ми можемо обходити спочатку ліве під-дерево, потім праве (послідовний обхід) і зворотний обхід. Вузол може бути поміщений у дерево двійкового пошуку тільки в якості кінцевого вузла. Якщо дерево є порожнім, то створюється новий екземпляр класу вузол дерева й вузол міститься в дереві. Якщо дерево не є порожнім, то програма порівнює, що вставляється в дерево значення зі значенням у кореневому вузлі і якщо менше, те поміщає в ліві під-дерева, а якщо більше, те в праві. Якщо значення рівні, то виводиться повідомлення, що повтор і не вставляється.

Задача 2.5. Використовуючи ієрархію й композицію класів, створити бінарне дерево. У бінарного дерева є кореневий вузол. Ми можемо вставляти вузол. Ми можемо обходити завширшки й зворотний обхід. Вузол може бути поміщений у дерево двійкового пошуку тільки в якості кінцевого вузла. Якщо дерево є порожнім, то створюється новий екземпляр класу вузол дерева й вузол міститься в дереві. Якщо дерево не є порожнім, то програма порівнює, що вставляється в дерево значення зі значенням у кореневому вузлі і якщо менше, те поміщає в ліві під-дерева, а якщо більше, те в праві. Якщо значення рівні, то виводиться повідомлення, що повтор і не вставляється.

Задача 2.6. Створити карту й використовую композицію - колоду карт. Конструктори колоди повинні ініціалізувати колоду впорядковане й випадковим образом. Створити похідний клас від колоди - пасьянс, у якому вибираються по три карти й, якщо дві крайні одного кольору, то їх викидають. Усю колоду проходять три рази.

Задача 2.7. Створити класи колесо, велосипед і автомобіль. Скласти з них ієрархію або композицію.

Задача 2.8. Використовуючи ієрархію й успадкування, створити класи вікна, вікна із заголовком і вікна із кнопкою.

Задача 2.9. Створити клас жорсткий диск, що має обсяг (Мбайт). Визначити конструктори й метод доступу. Створити клас комп'ютер, що містить клас жорсткий диск. Додатково є марка (вказівник на рядок), ціна. Визначити конструктори й деструктор. Визначити private-, public- похідний клас комп'ютерів з монітором розмір, що має додатково, монітора. Визначити конструктори, деструктори й функцію друку.

Задача 2.10. Створити клас процесор, що має потужність (Мгц). Визначити конструктори й метод доступу. Створити клас комп'ютер, що містить клас процесор. Додатково є марка (вказівник на рядок), ціна. Визначити конструктори й деструктор. Визначити private-, public- похідний клас комп'ютерів з монітором розмір, що має додатково, монітора. Визначити конструктори, деструктори й функцію друку.

Задача 2.11. Створити клас кнопка, що має розмір. Визначити конструктори й метод доступу. Створити клас вікно, що містить клас кнопка. Додатково є координати вікна. Визначити конструктори й деструктор. Визначити public- похідний клас вікно із кнопкою меню, що й має (вказівник на рядок). Визначити конструктори, деструктори й функцію друку.

Задача 2.12. Створити клас колесо, що має радіус. Визначити конструктори й метод доступу. Створити клас машин, що містить клас колесо. Додатково є марка (вказівник на рядок), ціна. Визначити конструктори й деструктор. Визначити public-похідний клас вантажівка, що має додатково вантажопідйомність. Визначити конструктори, деструктори й функцію друку.

Задача 2.13. Використовуючи ієрархію й композицію класів, створити бінарне дерево. У бінарного дерева є кореневий вузол. Ми можемо вставляти вузол. Ми можемо обходити завширшки.

Задача 2.14. Створити клас вікно, використовуючи композицію й ієрархію, визначити вікно із заголовком і кнопкою. Клас заголовок і кнопка теж повинні бути визначені. Навісити на кнопку функцію зворотного виклику - (метод класу вікна). Ця функція викликається, коли натискається будь-яка клавіша.

Завдання 3. Варіанти задач. Конструктори копіювання та оператор присвоювання при успадкуванні.

Задача 3.1. Створити ієрархію класів спортивна гра й футбол. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.2. Створити ієрархію класів утвір і вищий освіта. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.3. Створити ієрархію класів меблі й стіл. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.4. Створити клас рядків і похідний від нього - клас рядків із цифр. Визначити конструктори й деструктори, перевизначити вивід і введення в потік. Перевантажити оператор присвоювання й конструктор копіювання в базовому й похідному класі.

Задача 3.5. Створити клас рядків і похідний від нього - клас рядків із цифр. Визначити конструктори й деструктори, перевизначити вивід і введення в потік. Перевантажити оператор присвоювання й конструктор копіювання в базовому й похідному класі.

Задача 3.6. Створити ієрархію класів вектор і безпечний вектор з перевіркою виходу за межі. Безпечний вектор визначає змінні нижня й верхня межа. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.7. Створити ієрархію класів людей і викладач. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.8. Створити ієрархію класів студент і студент дипломник. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.9. Створити ієрархію класів чотирикутник і квадрат. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.10. Створити ієрархію класів людей і службовець. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.11. Створити ієрархію класів людей і студент. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.12. Створити ієрархію класів вікно й вікно із заголовком. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.13. Створити ієрархію класів крапка й кольорова крапка. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Задача 3.14. Створити ієрархію класів рядків і рядків із прописних букв. Перевизначити вивід у потік і введення з потоку, конструктор копіювання, оператор присвоювання через відповідні функції базового класу.

Завдання 4. Варіанти задач. Віртуальні функції.

Задача 4.1. Створити абстрактний базовий клас із віртуальною функцією - площа. Створити похідні класи: прямокутник, коло, прямокутний трикутник, трапеція зі своїми функціями площі. Для перевірки визначити масив посилань на абстрактний клас, яким присвоюється адреси різних об'єктів. Площа трапеції: $S=(a+b)h/2$.

Задача 4.2. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.

Задача 4.3. Створити абстрактний клас із віртуальною функцією: норма. Створити похідні класи: комплексні числа, вектор з 10 елементів, матриця (2x2). Визначити функцію норми - для комплексних чисел - модуль у квадраті, для вектора - корінь квадратний із суми елементів по модулю, для матриці - максимальне значення по модулю.

Задача 4.4. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки..

Задача 4.5. Створити абстрактний клас (криві) обчислення координати у для деякої х. Створити похідні класи: пряма, еліпс, гіпербола зі своїми функціями обчислення у залежно від вхідного параметра х. Рівняння прямої: $y=ax+b$, еліпса: $x^2/a^2+y^2/b^2=1$, гіперболи: $x^2/a^2-y^2/b^2=1$.

Задача 4.6. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.

Задача 4.7. Створити абстрактний базовий клас із віртуальною функцією - сума прогресії. Створити похідні класи: арифметична прогресія й геометрична прогресія. Кожний клас має два поля типу double. Перше - перший член прогресії, друге (double) - постійна різниця (для арифметичної) і постійне відношення (для геометричної). Визначити функцію обчислення суми, де параметром є кількість елементів прогресії. (Арифметична прогресія $a_j=a_0+jd$, $j=0,1,2,\dots$ Сума арифметичної прогресії: $s_n=(n+1)(a_0+a_n)/2$, Геометрична прогресія: $a_j=a_0r^j$, $j=0,1,2,\dots$ Сума геометричної прогресії: $s_n=(a_0-anr)/(1-r)$).

Задача 4.8. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.

Задача 4.9. Створити базовий клас список. Реалізувати на базі списку стек і черга з віртуальними функціями вставки й витаскування.

Задача 4.10. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.

Задача 4.11. Створити базовий клас - фігура, і похідні клас - коло, прямокутник, трапеція. Визначити віртуальні функції площа, периметр і вивід на печатку.

Задача 4.12. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.

Задача 4.13. Створити базовий клас - працівник і похідні класи - службовець із погодинною оплатою, службовець у штаті й службовець із процентною ставкою. Визначити функцію нарахування зарплати.

- Задача 4.14. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.
- Задача 4.15. Створити абстрактний базовий клас із віртуальною функцією - площа поверхні. Створити похідні класи: паралелепіпед, тетраедр, куля зі своїми функціями площі поверхні. Для перевірки визначити масив посилань на абстрактний клас, яким присвоюється адреси різних об'єктів. (Площа поверхні паралелепіпеда: $S=6xy$. Площа поверхні кулі: $S=4\pi r^2$. Площа поверхні тетраедра: $S=a^2\pi^3$)
- Задача 4.16. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.
- Задача 4.17. Створити клас людей, похідні від якого дівчини й молоді люди. Визначити віртуальну функцію реакції людину на знову побаченого іншу людину.
- Задача 4.18. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.
- Задача 4.19. Створити абстрактний базовий клас із віртуальною функцією - обсяг. Створити похідні класи: паралелепіпед, піраміда, тетраедр, куля зі своїми функціями обсягу. Для перевірки визначити масив посилань на абстрактний клас, яким присвоюється адреси різних об'єктів. (Об'єм паралелепіпеда - $V=xyz$ (x, y, z - сторони, піраміди: $V=xyh$ (x, y - сторони, h - висота), тетраедра: $V=\frac{a^3\pi^2}{12}$, кулі: $V=\frac{4\pi r^3}{3}$.)
- Задача 4.20. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.
- Задача 4.21. Створити абстрактний клас - ссавці. Визначити похідні класи - тварини й люди. У тварин визначити похідні класи собак і корів. Визначити віртуальні функції опису людину, собаки й корови.
- Задача 4.22. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.
- Задача 4.23. Створити базовий клас - Предок, у якого є ім'я. визначити віртуальну функцію друку. Створити похідний клас Дитина, у якій функція друку додатково виводить ім'я. Створити похідний клас від останнього класу - Онук, у якого є по батькові. Написати свою функцію друку.
- Задача 4.24. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.
- Задача 4.25. Створити клас живучих з місцем розташування. Визначити наслідувані класи - лиса, кролик і трава. Лисиця їсть кролика. Кролик їсть траву. Лисиця може вмерти - визначений вік. Кролик теж може вмерти. Крім цього визначений клас - відсутність життя. Якщо в околиці є більше трави, чому кроликів, то трава залишається, інакше трава з'їдена. Якщо лис занадто старий він може вмерти. Якщо лис занадто багато (більше 5 в околиці), лиси більше не з'являються. Якщо кроликів менше лисиць, то лис їсть кролика.
- Задача 4.26. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.
- Задача 4.27. Створити абстрактний базовий клас із віртуальною функцією - коріння рівняння. Створити похідні класи: клас лінійних рівнянь і клас квадратних рівнянь. Визначити функцію обчислення корінь рівнянь.

Задача 4.28. Створити клас - дані - абстрактний базовий клас. Створити похідні класи - дані типу сигнал, дані типу результат обробки й допоміжні дані. Усі дані мають функції відображення, збереження й обробки.

Завдання 5. Варіанти задач. Множинне успадкування.

Задача 5.1. Створити ієрархію типів, що описує - студента, батька сімейства й студента-батька сімейства. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.

Задача 5.2. Створити ієрархію типів, що описує - людину, студента, батька сімейства й студента-батька сімейства. Використовувати віртуальні базові класи.

Задача 5.3. Створити ієрархію типів, що описує працівника й батька-сімейства, і працівника-батька сімейства. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.

Задача 5.4. Створити ієрархію типів, що описує - людину, працівника, батька сімейства й працівника-батька сімейства. Використовувати віртуальні базові класи.

Задача 5.5. Створити ієрархію типів - файл для читання, файл для запису й файл для читання й запису. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.

Задача 5.6. Створити ієрархію типів, що описує файл, файл для читання, файл для запису й файл для читання й запису. Використовувати віртуальні базові класи.

Задача 5.7. Створити ієрархію типів, що описує працівника й жінку, і працівника-жінку сімейства. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.

Задача 5.8. Створити ієрархію типів, що описує людину, працівника й жінку, і працівника-жінку сімейства. Використовувати віртуальні базові класи.

Задача 5.9. Створити ієрархію типів, що описує операційну систему й прикладне програмне забезпечення, і Windows NT як операційну систему й прикладне програмне забезпечення. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.

Задача 5.10. Створити ієрархію типів, що описує програмне забезпечення, операційну систему й прикладне програмне забезпечення, і Windows NT як операційну систему й прикладне програмне забезпечення. Використовувати віртуальні базові класи.

Задача 5.11. Створити ієрархію типів, що описує дані - сигнал, дані результат обробки й дані, як результат обробки сигналу, що й представляють собою сигнал. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.

Задача 5.12. Створити ієрархію типів, що описує дані - сигнал, дані результат обробки й дані, як результат обробки сигналу, що й представляють собою сигнал. Використовувати віртуальні базові класи.

Задача 5.13. Створити ієрархію типів - море, заливши й бухта. . Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.

Задача 5.14. Створити ієрархію типів - солена вода, море, заливши й бухта. Використовувати віртуальні базові класи.

Задача 5.15. Створити ієрархію типів - корабель, пасажирський транспорт і пасажирський корабель. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.

- Задача 5.16. Створити ієрархію типів - транспорт, корабель, пасажирський транспорт і пасажирський корабель. Використовувати віртуальні базові класи.
- Задача 5.17. Створити ієрархію типів - машина, пасажирський транспорт і автобус. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.
- Задача 5.18. Створити ієрархію типів - машина, пасажирський транспорт і автобус. Використовувати віртуальні базові класи.
- Задача 5.19. Створити ієрархію типів, що описує - студента, батька сімейства й студента-батька сімейства. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.
- Задача 5.20. Створити ієрархію типів, що описує - людину, студента, батька сімейства й студента-батька сімейства. Використовувати віртуальні базові класи.
- Задача 5.21. Створити ієрархію типів, що описує працівника й батька-сімейства, і працівника-батька сімейства. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.
- Задача 5.22. Створити ієрархію типів, що описує - людину, працівника, батька сімейства й працівника-батька сімейства. Використовувати віртуальні базові класи.
- Задача 5.23. Створити ієрархію типів - файл для читання, файл для запису й файл для читання й запису. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.
- Задача 5.24. Створити ієрархію типів, що описує файл, файл для читання, файл для запису й файл для читання й запису. Використовувати віртуальні базові класи.
- Задача 5.25. Створити ієрархію типів - корабель, пасажирський транспорт і пасажирський корабель. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.
- Задача 5.26. Створити ієрархію типів - транспорт, корабель, пасажирський транспорт і пасажирський корабель. Використовувати віртуальні базові класи.
- Задача 5.27. Створити ієрархію типів, що описує операційну систему й прикладне програмне забезпечення, і Unix як операційну систему й прикладне програмне забезпечення. Класи повинні конструктори, включаючи конструктор копіювання, віртуальні деструктори, перевантажені функції виводу в потік і введення в потік.
- Задача 5.28. Створити ієрархію типів, що описує програмне забезпечення, операційну систему й прикладне програмне забезпечення, і Unix як операційну систему й прикладне програмне забезпечення. Використовувати віртуальні базові класи.
- Задача 5.29.