

Лабораторна робота №23

Тема: ООП. Вступ до ООП

Мета: Навчитися використовувати ООП.

Індивідуальне завдання

Загальне завдання.

Для предметної галузі з розділу “Розрахункове завдання / Індивідуальні завдання” розробити два класи: - клас, що відображає сутність “базового класу”, у тому числі: - конструктор за замовчуванням, копіювання та конструктор з аргументами (реалізація конструкторів повинна бути продемонстрована за допомогою списків ініціалізацій); - деструктор; - клас, що має у собі динамічний масив об’єктів базового класу та має в собі методи додавання, видалення елемента, отримання елемента по індексу (або ідентифікатору), вивід усіх елементів на екран. Рекомендовані сигнатури методів: - додавання:

```
void CList::addPhone(Phone& phone);
```

- видалення:

```
void CList::removePhone(int index);
```

- отримання по індексу:

```
CPhone& CList::getPhone(int index);
```

- виведення усіх елементів (при цьому цей метод повинен викликати метод `CList::getPhone(int index)`, щоб не було дублювання коду):

```
void CList::showAll();
```

- метод 1 обходу колекції. Приклад сигнатури такого методу (У наведеному прикладі реалізоване завдання пошуку самого дешевого телефону з заданою діагоналлю (повертається один телефон):

```
CPhone& findCheapestPhone(float diagonal);
```

Додаткові умови виконання завдання:

- усі поля “базового класу” повинні бути приватними та мати публічні гетери та сетери (модифікатори доступу), використовувати механізм інкапсуляції;
- усі функції, що не повинні змінювати поля поточного об’єкта, повинні бути константними;
- усі аргументи функцій, що не змінюються, по можливості також повинні бути константними. Якщо їх не можна зробити константними, у такому разі повинно бути обґрунтування цього;
- у класі-списку метод додавання елемента не повинен вводити дані з клавіатури або файлу, а повинен приймати вже готовий об’єкт для додавання. Метод вводу даних має бути відокремленим;
- продемонструвати відсутність витоків пам’яті;
- продемонструвати роботу розроблених методів класу-списку за допомогою модульних тестів.

Хід роботи

1. Написання коду для базового класу, класу списку та класу помічника.

```
class Pupils
{
private:
    int numberSchool;
    string fullname;
    int age;
    string characteristic;
    int grade;
    bool insurance;
};
```

Рисунок 1.1 - Поля базового класу.

```
public:
    Pupils();
    Pupils(const int numberSchool, const string fullname, const int age, const string characteristic, const int grade, const bool insurance);
    Pupils(const Pupils& obj);
    ~Pupils();

    Pupils& operator=(const Pupils& obj);
    friend bool operator==(const Pupils& obj1, const Pupils& obj2) { ... }
    friend bool operator!=(const Pupils& obj1, const Pupils& obj2) { ... }
    friend std::ostream& operator<< (std::ostream& out, const Pupils& obj) {
        out << "\nNumberSchool:" << obj.numberSchool;
        out << "\nFullName:" << obj.fullname;
        out << "\nAge:" << obj.age;
        out << "\nCharacteristic:" << obj.characteristic;
        out << "\nGrade" << obj.grade;
        out << "\nInsurance:" << obj.insurance ? "Yes" : "No";
        return out;
    }
    friend std::istream& operator>> (std::istream& in, Pupils& obj) {
        in >> obj.numberSchool;
        in >> obj.fullname;
        in >> obj.age;
        in >> obj.characteristic;
        in >> obj.grade;
        in >> obj.insurance;

        return in;
    }

    string printPupils();

    void setNumberSchool(int numberSchool);
    void setFullName(string fullName);
    void setAge(int age);
    void setCharac(string charac);
    void setGrade(int grade);
    void setInsurance(bool insurance);

    int getNumberSchool();
    string getFullName();
    int getAge();
    string getCharac();
    int getGrade();
    bool getInsurance();
};
```

Рисунок 1.2 - Конструктор копіювання зі списком ініціалізації.

```

class List
{
private:
    std::vector<Pupils> list;

public:
    void addPupils(Pupils& pupils);
    void addObject();
    void removePupils(const int index);
    void showAll();

    int getSize();
    Pupils& getPupils(const int index);
    Pupils& getPupilsGrade(const int grade);
    string check(regex reg);

    Pupils& operator[](const int index);
    friend std::ostream& operator << (std::ostream& out, const List& obj) {
        for (int i = 0; i < obj.list.size(); i++)
            out << obj.list[i];

        return out;
    }

    void readFromFile();
    void writeToFile();
};

```

Рисунок 1.3 - Для динамічного масиву було використано бібліотеку vector, на рисунку зображені методи для роботи з класом списком.

```

Pupils::Pupils(const Pupils& obj) :numberSchool(obj.numberSchool), fullname(obj.fullname), age(obj.age),
characteristic(obj.characteristic), grade(obj.grade), insurance(obj.insurance)
{
    cout << "\n\t\tThere was a COPY constructor here\n";
}

```

Рисунок 1.4 - Конструктор копіювання зі списком ініціалізації..

```

A list of pupils.Select an action from the following.
1.Write list to screen.
2.Generate and add pupils.
3.Delete pupils by index.
4.Search for pupils with insurance.
5.Reading pupils from a file.
6.Writing pupils in file.
0.Exit.
Your choice:

```

Рисунок 1.5 - Початкове меню.

```
Your choice: 2

There was a DEFAULT constructor here

Create a new pupils now.
  Enter full name : Sam
  Enter age : 15
  Enter number school : 598
  Describe the positive qualities in 3 words : Creativity, Smart
Try again :Creativity,Smart
  Enter grade : 9
  Availability of insurance (1 - yes, 0 - no) : 1
```

Рисунок 1.6 - Приклад створення та додавання елементу у список.

```
Your choice: 1

NumberSchool:152
Fullname:Alex
Age:15
Characteristic:Cool,Energy
Grade:9
Insurance:0

NumberSchool:758
Fullname:Max
Age:13
Characteristic:Cool
Grade:5
Insurance:0

NumberSchool:254
Fullname:Andrew
Age:18
Characteristic:Cool
Grade:11
Insurance:0

NumberSchool:256
Fullname:Julia
Age:18
Characteristic:Smart
Grade:11
Insurance:0

NumberSchool:598
Fullname:Sam
Age:15
Characteristic:Creativity,Smart
Grade:9
Insurance:1

Successful.
```

Рисунок 1.7 - Приклад використання методу для виводу всіх елементів.

A list of pupils.Select an action from the following.

- 1.Write list to screen.
- 2.Generate and add pupils.
- 3.Delete pupils by index.
- 4.Search for pupils with insurance.
- 5.Reading pupils from a file.
- 6.Writing pupils in file.
- 0.Exit.

Your choice: 3

Choose index(0-5):1

There was DESTRUCTOR here

Successful.

A list of pupils.Select an action from the following.

- 1.Write list to screen.
- 2.Generate and add pupils.
- 3.Delete pupils by index.
- 4.Search for pupils with insurance.
- 5.Reading pupils from a file.
- 6.Writing pupils in file.
- 0.Exit.

Your choice: 1

NumberSchool:758

Fullname:Max

Age:13

Characteristic:Cool

Grade:5

Insurance:0

NumberSchool:254

Fullname:Andrew

Age:18

Characteristic:Cool

Grade:11

Insurance:0

Рисунок 1.8 - Приклад використання методу видалення елемента зі списку.

```
A list of pupils.Select an action from the following.

1.Write list to screen.
2.Generate and add pupils.
3.Delete pupils by index.
4.Search for pupils with insurance.
5.Reading pupils from a file.
6.Writing pupils in file.
0.Exit.

Your choice: 4

NumberSchool:598
Fullname:
Age:15
Characteristic:
Grade:9
Insurance:1

Successful.
```

Рисунок 1.9 - Приклад використання методу пошуку учнів зі страхуванням.

1. Write list to screen.
2. Generate and add pupils.
3. Delete pupils by index.
4. Search for pupils with insurance.
5. Reading pupils from a file.
6. Writing pupils in file.
0. Exit.

Your choice: 5

There was DESTRUCTOR here

There was DESTRUCTOR here

There was DESTRUCTOR here

Here was a constructor with PARAMETERS

There was a COPY constructor here

There was DESTRUCTOR here

Here was a constructor with PARAMETERS

There was a COPY constructor here

There was DESTRUCTOR here

Here was a constructor with PARAMETERS

There was a COPY constructor here

There was DESTRUCTOR here

Here was a constructor with PARAMETERS

There was a COPY constructor here

There was DESTRUCTOR here

Here was a constructor with PARAMETERS

There was a COPY constructor here

Рисунок 1.10 - Приклад зчитування з файлу

```
Your choice: 1

NumberSchool:152
Fullname:Alex
Age:15
Characteristic:Cool,Energy
Grade:9
Insurance:0

NumberSchool:758
Fullname:Max
Age:13
Characteristic:Cool
Grade:5
Insurance:0

NumberSchool:254
Fullname:Andrew
Age:18
Characteristic:Cool
Grade:11
Insurance:0

NumberSchool:256
Fullname:Julia
Age:18
Characteristic:Smart
Grade:11
Insurance:0

NumberSchool:598
Fullname:Sam
Age:15
Characteristic:Creativity,Smart
Grade:9
```

Рисунок 1.11 - Вміст списку після зчитування

Your choice: 6

Successful.

A list of pupils. Select an action from the following

1. Write list to screen.
2. Generate and add pupils.
3. Delete pupils by index.
4. Search for pupils with insurance.
5. Reading pupils from a file.
6. Writing pupils in file.
0. Exit.

Your choice: 1

NumberSchool:152
Fullname:Alex
Age:15
Characteristic:Cool,Energy
Grade:9
Insurance:0

NumberSchool:758
Fullname:Max
Age:13
Characteristic:Cool
Grade:5
Insurance:0

NumberSchool:254
Fullname:Andrew
Age:18
Characteristic:Cool
Grade:11
Insurance:0

NumberSchool:256
Fullname:Julia
Age:18
Characteristic:Smart
Grade:11
Insurance:0

Рисунок 1.12 - Вміст консолі і запис у файл


 *text.txt – Блокнот
Файл Правка Формат Вид Справка
152 Alex 15 Cool,Energy 9 0
758 Max 13 Cool 5 0
254 Andrew 18 Cool 11 0
256 Julia 18 Smart 11 0
598 Sam 15 Creativity,Smart 9 1|

Рисунок 1.12 - Вміст файлу після запису

Висновок: Навчилися використовувати ООП.