

Автор: Татаренко А., КІТ-119а

Дата: 8 січня 2019

## Лабораторна робота №1.

### Класи

**Тема роботи:** Класи та специфікатори доступу. Інкапсуляція. Константи.

**Мета роботи:** отримати базові знання про класи. Дослідити механізм інкапсуляції.

### 1 Завдання до роботи

#### Індивідуальне завдання 19.

Для предметної галузі з таблиці розробити два класи:

- клас, що відображає сутність «базового класу». При цьому, в даному класі повинно бути мінімум три числових поля (бажано, щоб одне з цих полів було унікальним ідентифікатором об'єкта);
- 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.

### 2 Розробка алгоритму розв'язання задачі.

#### 2.1 Опис змінних

```
Arr stud_array; class Student; class Arr;
```

Класи, методи, функції

### 3 Код програми

#### basic\_class.h:

```
#define _CRTDBG_MAP_ALLOC
#include <stdlib.h>
#include <crtdbg.h>

#include <iostream>
#define El 5

using std::cout;
using std::cin;
using std::endl;

class Student
{
private:

    int age;
    int number_stud;
    int middle_mark;

public:

    void set_number_stud(int);
    int get_number_stud() const;

    void set_age(int);
    int get_age() const;

    void set_middle_mark();
    int get_middle_mark() const;

};
```

#### class\_list.h:

```
#pragma once
#include "basic_class.h"

class Arr
{
private:

    Student* array_stud;

    int count = El;

public:

    void create_array();

    void print_array() const;

    void delete_one();

    void add();

    void select() const;

    void fill_array();
```

```
    int count_plus();

    int count_minus();

    void print_one(Student) const;

    void delete_array();

};
```

### menu.h:

```
#pragma once
#include "class_list.h"

void menu(Arr);
```

### test.h:

```
#pragma once
#include "class_list.h"

bool test_count_plus(Arr);
bool test_count_minus(Arr);
```

### basic\_class.cpp:

```
#include "basic_class.h"

int Student::get_number_stud() const
{
    return number_stud;
}

void Student::set_number_stud(int n)
{
    number_stud = n;
}

int Student::get_age() const
{
    return age;
}

void Student::set_age(int a)
{
    age = a;
}

int Student::get_middle_mark() const
{
    return middle_mark;
}

void Student::set_middle_mark()
{
    middle_mark = rand() % 10;
}
```

## class\_list.cpp:

```
#include "class_list.h"

void Arr::create_array()
{
    array_stud = new Student[count];
}

void Arr::fill_array()
{
    for (int i = 0; i < count; i++)
    {
        array_stud[i].set_number_stud(i + 1);
        array_stud[i].set_age(i + 17);
        array_stud[i].set_middle_mark();
    }
}

void Arr::print_array() const
{
    for(int i = 0; i < 50; i++)
        cout << "-";
    cout << endl;
    ////////////////
    for (int i = 0; i < count; i++)
        cout << "Number: " << array_stud[i].get_number_stud() << "\tAge: " <<
array_stud[i].get_age() << "\t\tMiddle mark: " << array_stud[i].get_middle_mark() <<
endl;
    ////////////////
    for (int i = 0; i < 50; i++)
        cout << "-";
    cout << endl;
}

void Arr::delete_one()
{
    int input, i_arr = 0;
    cout << "Enter a number: ";
    cin >> input;

    for (int i = 0; i < El; i++)
        if (input == array_stud[i].get_number_stud())
        {
            count_minus();

            Student* array_stud_new = new Student[count];

            for (int i = 0; i < count + 1; i++)
            {
                if (input == array_stud[i].get_number_stud())
                    continue;
                array_stud_new[i_arr++] = array_stud[i];
            }

            delete[] array_stud;

            array_stud = new Student[count];

            for (int i = 0; i < count; i++)
            {
                array_stud[i] = array_stud_new[i];
            }
        }
}
```

```

        delete[] array_stud_new;

        print_array();

        return;
    }
    ////////////
    for (int i = 0; i < 50; i++)
        cout << "-";
    cout << endl;
    ////////////
    cout << "That student is not exist!" << endl;
    ////////////
    for (int i = 0; i < 50; i++)
        cout << "-";
    cout << endl;
    ////////////
}

void Arr::add()
{
    count_plus();

    Student* array_stud_new = new Student[count];

    int i = 0;

    for (i; i < count - 1; i++)
    {
        array_stud_new[i] = array_stud[i];
    }

    array_stud_new[count - 1].set_number_stud(i + 1);
    array_stud_new[count - 1].set_age(i + 17);
    array_stud_new[count - 1].set_middle_mark();

    delete[] array_stud;

    array_stud = new Student[count];

    for (i = 0; i < count; i++)
    {
        array_stud[i] = array_stud_new[i];
    }

    delete[] array_stud_new;

    print_array();
}

int Arr::count_plus()
{
    count = count + 1;

    return count;
}

int Arr::count_minus()
{
    count = count - 1;

    return count;
}

```

```

void Arr::select() const
{
    int input = 0;
    cout << "Enter a number: ";
    cin >> input;

    for (int i = 0; i < E1; i++)
    {
        if (input == array_stud[i].get_number_stud())
        {
            print_one(array_stud[i]);
            return;
        }
    }
    ////////////
    for (int i = 0; i < 50; i++)
        cout << "-";
    cout << endl;
    ////////////
    cout << "That student is not exist!" << endl;
    ////////////
    for (int i = 0; i < 50; i++)
        cout << "-";
    cout << endl;
    ////////////
}

void Arr::print_one(Student stud) const
{
    for (int i = 0; i < 50; i++)
        cout << "-";
    cout << endl;
    ////////////
    cout << "Numder: " << stud.get_number_stud() << "\tAge: " << stud.get_age() <<
"\t\tMiddle mark: " << stud.get_middle_mark() << endl;
    ////////////
    for (int i = 0; i < 50; i++)
        cout << "-";
    cout << endl;
}

void Arr::delete_array()
{
    delete[] array_stud;
}

```

## main.cpp:

```

/**
 * @mainpage
 * <b> Лабораторна робота № 1. <br/> Класи </b>
 * <br/><b><i>Мета роботи:</i></b>: отримати базові знання про класи. Дослідити механізм
інкапсуляції. <br/>
 * <b><i>Індивідуальне завдання 19:</i></b>
 * Для предметної галузі розробити два класи. <br/>
 * <br/>
 * <br/> <b> Arr stud_array; class Student; class Arr; </b> Класи, методи, функції </b>
 * @author Tatarenko A.
 * @date 4-mar-2020
 * @version 1.0
 */

#include "menu.h"

```

```

int main()
{
    Arr stud_array;

    int input = 0;

    cout << "Enter 1 to create array, 2 to exit: ";
    cin >> input;

    if (input == 1)
    {
        stud_array.create_array();
        stud_array.fill_array();
        stud_array.print_array();

        menu(stud_array);

        stud_array.delete_array();
    }

    //////////////////////////////////////
    int l = _CrtDumpMemoryLeaks(); // Контроль витоку пам'яті
    if (l == 0)
        cout << "No memory leaks!" << endl;
    else
        cout << "Memory leaks!" << endl;

    system("PAUSE");
    return 0;
}

```

### menu.cpp:

```

#include "menu.h"

void menu(Arr stud_array)
{
    int input = 0;

    cout << "1) Add object at the end;" << endl << "2) Delete object;" << endl << "3)
Select;" << endl << "4) Show all objects;" << endl << "5) Exit;" << endl << "Your choice:
";
    cin >> input;

    switch (input)
    {
    case 1:
        stud_array.add();
        break;
    case 2:
        stud_array.delete_one();
        break;
    case 3:
        stud_array.select();
        break;
    case 4:
        stud_array.print_array();
        break;
    case 5:
        return;
    default:
        break;
    }
}

```

```
        menu(stud_array);
    }
}
```

## test.cpp:

```
#include "test.h"

int main()
{
    Arr stud_array;

    if (test_count_plus(stud_array))
        cout << "Function count_plus is right!" << endl;
    else
        cout << "Function count_plus is not right!" << endl;

    if (test_count_minus(stud_array))
        cout << "Function count_minus is right!" << endl;
    else
        cout << "Function count_minus is not right!" << endl;

    system("PAUSE");
}

bool test_count_plus(Arr p)
{
    int exp = E1 + 1;
    int cnt = p.count_plus();

    if (cnt == exp)
        return 1;
    else
        return 0;
}

bool test_count_minus(Arr p)
{
    int exp = E1 - 1;
    int cnt = p.count_minus();

    if (cnt == exp)
        return 1;
    else
        return 0;
}
```



## 4 Результати тестування

```
Enter 1 to create array, 2 to exit: 1
-----
Nunder: 1      Age: 17      Middle mark: 1
Nunder: 2      Age: 18      Middle mark: 7
Nunder: 3      Age: 19      Middle mark: 4
Nunder: 4      Age: 20      Middle mark: 0
Nunder: 5      Age: 21      Middle mark: 9
-----
1) Add object at the end;
2) Delete object;
3) Select;
4) Show all objects;
5) Exit;
Your choice:
```

## 5 Опис результатів

Програма виводить список студентів. Виконує видалення, додавання, вибір об'єкта.

Висновок:

Протягом цієї роботи було отримано базові знання про класи. Досліджено механізм інкапсуляції.