

ЗВІТ

Практична робота №14

з дисципліни: Об'єктно-орієнтоване програмування

Студент: Вівчар Вадим Вікторович

Група: АЛК-43

## ТЕОРЕТИЧНІ ВІДОМОСТІ

Об'єктно-орієнтоване програмування базується на принципах:

- Інкапсуляція
- Наслідування
- Поліморфізм
- Абстракція

У даній роботі розглянуто дружні функції, делегування, заборону копіювання та асоціювання класів Employer, Employee, Accountant, Person, Position.

## ХІД ВИКОНАННЯ РОБОТИ

КРОК 1. Створення класів Person, Position, Employee, Accountant, Employer.

Було реалізовано інкапсуляцію, дружні функції, асоціації та делегування.

КРОК 2. Початковий стан програми — всі позиції VACANT.

КРОК 3. Найм двох працівників — Ivan Ivanov та Petro Petrov.

КРОК 4. Спроба найняти третього працівника — вакансій немає.

КРОК 5. Звільнення працівника на позиції 0.

КРОК 6. Повторна спроба звільнення — обробка помилки.

КРОК 7. Робота утиліти Calc + exe(): операції +, ^, %, =, n.

## РИСУНОК 1 — Найм, звільнення, відсутність вакансій

```
Консоль отладки Microsoft Visual Studio
Position index 1: VACANT
=====

Hiring two employees...
Trying to hire Ivan Ivanov to position Developer
Employee Ivan Ivanov takes position Developer
Hired Ivan Ivanov on position index 0
Trying to hire Petro Petrov to position QA Engineer
Employee Petro Petrov takes position QA Engineer
Hired Petro Petrov on position index 1

=== STAFF STATE ===
Position index 0: OCCUPIED by Ivan Ivanov (Developer, salary 30000)
Position index 1: OCCUPIED by Petro Petrov (QA Engineer, salary 25000)
=====

Trying to hire third employee (no vacancies)...
Trying to hire Stepan Stepanych to position Developer
No vacant positions for Stepan Stepanych

=== STAFF STATE ===
Position index 0: OCCUPIED by Ivan Ivanov (Developer, salary 30000)
Position index 1: OCCUPIED by Petro Petrov (QA Engineer, salary 25000)
=====

Firing employee at position 0...
Trying to fire position 0
Employee Ivan Ivanov fired from Developer
Position 0 is now vacant.

=== STAFF STATE ===
Position index 0: VACANT
Position index 1: OCCUPIED by Petro Petrov (QA Engineer, salary 25000)
=====

Trying to fire already vacant position 0...
Trying to fire position 0
Position 0 is already vacant.

=== DEMO: Calc + exe() ===

Initial: a = 10, x = 3
After '+': a = 13, x = 3
After '^': a = 2197, x = 3
After '%': a = 1, x = 3
After '-': a = 3, x = 3
After 'n': a = 3, x = -3
Employee Boss with 2 positions was deleted
Employee Petro Petrov fired from QA Engineer
Position QA Engineer destroyed
Position Developer destroyed
Person Stepan Stepanych destroyed
Person Petro Petrov destroyed
Person Ivan Ivanov destroyed
Person Boss destroyed

D:\std\ConsoleApplication3\x64\Debug\ConsoleApplication3.exe (процесс 7960) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" -> "Параметры" -> "Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно.
```

РИСУНОК 2 — Завершения програми та калькулятор exe()

```
Консоль отладки Microsoft Visual Studio
Position index 0: VACANT
Position index 1: OCCUPIED by Petro Petrov (QA Engineer, salary 25000)
=====

Trying to fire already vacant position 0...
Trying to fire position 0
Position 0 is already vacant.

=== DEMO: Calc + exe() ===

Initial: a = 10, x = 3
After '+': a = 13, x = 3
After '^': a = 2197, x = 3
After '%': a = 1, x = 3
After '=': a = 3, x = 3
After 'n': a = 3, x = -3
Employer Boss with 2 positions was deleted
Employee Petro Petrov fired from QA Engineer
Position QA Engineer destroyed
Position Developer destroyed
Person Stepan Stepanych destroyed
Person Petro Petrov destroyed
Person Ivan Ivanov destroyed
Person Boss destroyed

D:\std\ConsoleApplication3\x64\Debug\ConsoleApplication3.exe (процесс 15104) завершил работу с кодом 0 (0x0).
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно: _
```

## Код Employer::hire()

```
void Employer::hire(const Person& person, const Position& position) {
    size_t i = 0;
    while (i < _volume && !_office[i]._free) i++;

    if (i == _volume) {
        cout << "No vacant positions for " << person.getName() << endl;
        return;
    }

    _office[i]._pemployee = new Employee(person, position);
    _office[i]._free = false;
}
```

## Код Employer::fire()

```
void Employer::fire(const size_t i) {  
    if (i >= _volume) return;  
  
    if (_office[i]._free) {  
        cout << "Position " << i << " is already vacant." << endl;  
        return;  
    }  
  
    delete _office[i]._pemployee;  
    _office[i]._free = true;  
}
```

Код утиліти exe()

```
double exe(Calc& c, char op) {  
    switch (op) {  
        case '+': return c.add();  
        case '^': return c._a = pow(c._a, c._x);  
        case '%': return c._a = fmod(c._a, c._x);  
        case '=': return c._a = c._x;  
        case 'n': return c._x = -c._x;  
        default: return c._a;  
    }  
}
```

## ВИСНОВОК

Під час виконання роботи було реалізовано:

- систему найму та звільнення працівників;
- інкапсуляцію, дружні функції та делегування;
- механізм перевірки вакансій та обробку помилок;
- утиліту `exe()` для виконання арифметичних операцій.

Мета роботи досягнута.