Звіт

Автор: Чугунов В.Ю

КІТ-119а

Лабораторна робота №5

**КЛАСИ. АГРЕГАЦІЯ. КОМПОЗИЦІЯ.  
КЛЮЧОВІ СЛОВА *typedef*****ТА *auto***

**Мета:** Порівняти поняття агрегації та композиції. Отримання знання про призначення ключових слів *typedef* та *auto*.

1 ЗАВДАННЯ ДО РОБОТИ

**Загальне завдання.** Дослідити заздалегідь визначені типи даних з бібліотеки <*cstddef*> / <*stddef.h*> Модернізувати розроблені у попередній роботі класи наступним чином:  
 - замінити типи даних, що використовуються при індексуванні на типи з вказаної бібліотеки;  
 - в базовий клас додати два поля, що мають кастомний тип даних, які будуть відображати відношення «агрегація» та «композиція».  
 - ввести використання ключового слова *auto* як специфікатора збері-гання типу змінної. Визначити плюси та мінуси цього використання.

2 РОЗРОБКА АЛГОРИТМІВ ФУНКЦІЙ

Створено 3 нових класи для відображення відношень «агрегації» та «композиції»:

Клас Aggregation, що містить у собі лише метод getInfo, який виводить у консоль повідомлення, що це є відношення «агрегація».

Клас ForTest, який містить у собі об’єкт класу Aggregation. Також він має свій метод getInfo, який просто викликає метод getInfo об’єкта класу Aggregation.

У класі Lessons створено клас Composition, що містить у собі лише метод getInfo, який виводить у консоль повідомлення, що це є відношення «композиція». Також додано 2 метода: getComposInfo та getAggregInfo, які викликають методи getInfo у класах Composition та Aggregation відповідно.

3 ТЕКСТ ПРОГРАМИ

**Текст файлу Lessons.cpp**

#include <string>

using std::string;

using std::cout;

using std::cin;

using std::endl;

#pragma once

class Aggregation

{

public:

void getInfo()

{

cout << "This is an aggregation relationship." << endl;

};

};

class Lessons

{

public:

Lessons();

Lessons(const string t, const string name, const string ID, const int time, const int qty);

Lessons(const Lessons & copyObject);

~Lessons();

void set\_audienceID(const string index);

void set\_lecturerName(const string name);

void set\_lessonType(const string type);

void set\_numOfWorks(const int number);

void set\_lessonStartTime(const int time);

const string get\_audienceID() const;

const string get\_lecturerName() const;

const string get\_lessonType() const;

const int get\_numOfWorks() const;

const int get\_lessonStartTime() const;

void generateLesson();

void getComposInfo() { compos.getInfo(); };

void getAggregInfo() { aggreg.getInfo(); };

private:

string audienceID;

string lecturerName;

string lessonType;

int numOfWorks;

int lessonStartTime;

Aggregation aggreg;

class Composition

{

public:

void getInfo()

{

cout << "This is a composition relationship." << endl;

};

} compos;

};

class ForTest

{

public:

void getInfo()

{

aggreg.getInfo();

}

private:

Aggregation aggreg;

};

ВИСНОВКИ

При виконанні лабораторної роботи з бібліотеки <*cstddef*> було взято тип даних *size\_t*, який було використано для створення змінних при індексуванні. Ключове слово *auto* було використано у циклах для зберігання типу змінної. З плюсів можна зазначити, що це полегшує написання коду, непотрібно пильно слідкувати за типом змінної. Але цей же плюс є і мінусом, таким чином можна забути якого саме типу була та чи інша змінна. Для даної роботи використання ключового слова *typedef* не потрібно.