

Міністерство освіти і науки України

Національний університет «Львівська політехніка»



Лабораторна робота № 5

з дисципліни: «Автоматизоване проектування комп'ютерних систем»,

на тему: «Впровадження тестів, що виконуються автоматично.»

Виконав:ст. гр. КІ-47

Ковальчук А.О.

Прийняв:

к.т.н. асит. каф. СКС

Кіцера А.О.

Львів – 2024

Task 5. Implement automated tests:

1. Implement or use existing test framework;
2. Create a set of automated tests;
3. Test report should contain number of all tests, passed tests, failed tests, coverage;
4. Coverage must be more than 80%
5. Required steps

Вариант 8:

Student number	Game	config format
8	rock paper scissors	JSON

Теоретичні відомості

Автоматизовані тести є невід'ємною частиною розробки програмного забезпечення, які виконуються автоматично для перевірки правильності роботи програм та виявлення можливих помилок. У мові програмування C++, для написання тестів, можна використовувати вбудовану бібліотеку `cassert` (`assert.h` у C), яка містить макрос `assert`. Цей макрос використовується для визначення виразів, які мають бути істинними. У випадку, якщо вираз виявиться хибним (`FALSE`), програма завершиться з повідомленням про помилку.

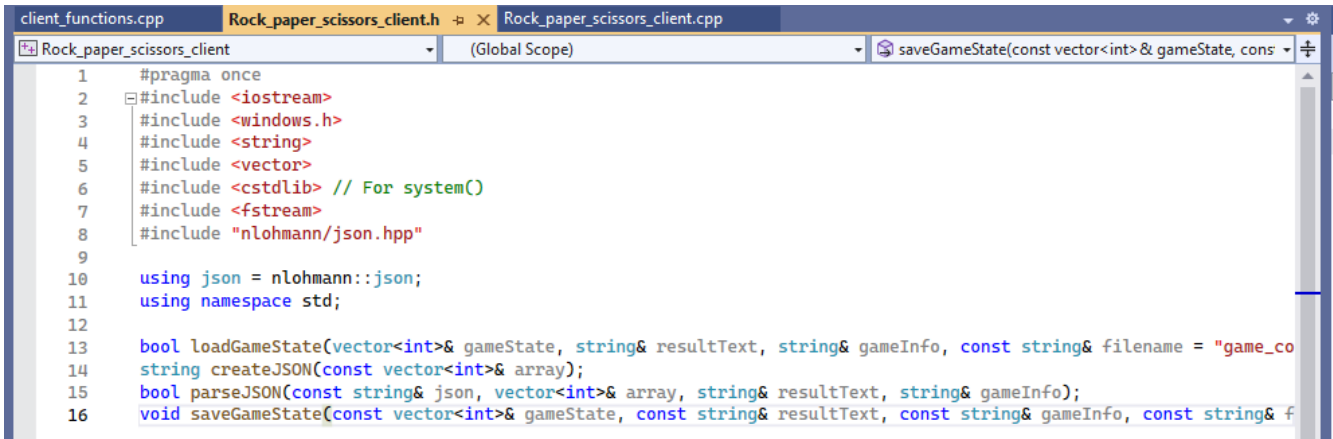
Для зроблення тестування більш систематичним та автоматизованим, розробники часто використовують спеціальні фреймворки для тестування, такі як `Google Test` або `Catch`. Ці фреймворки дозволяють створювати розширені тестові сценарії, вести звіти про результати тестування, та автоматизувати процес виявлення помилок.

Тести в розробці програмного забезпечення є важливою частиною процесу, оскільки вони дозволяють перевіряти правильність роботи коду, виявляти можливі помилки та забезпечувати стабільність програми. Тестування може бути використане на різних етапах розробки, від написання нового коду до внесення змін у вже існуючий.

Тестування може бути виконане як вручну (`manual testing`), так і автоматично (`automated testing`). Автоматизовані тести мають численні переваги, такі як швидкість виконання, можливість запуску тестів в будь-який момент, ідентифікація помилок на ранніх етапах розробки та можливість повторного використання тестових сценаріїв.

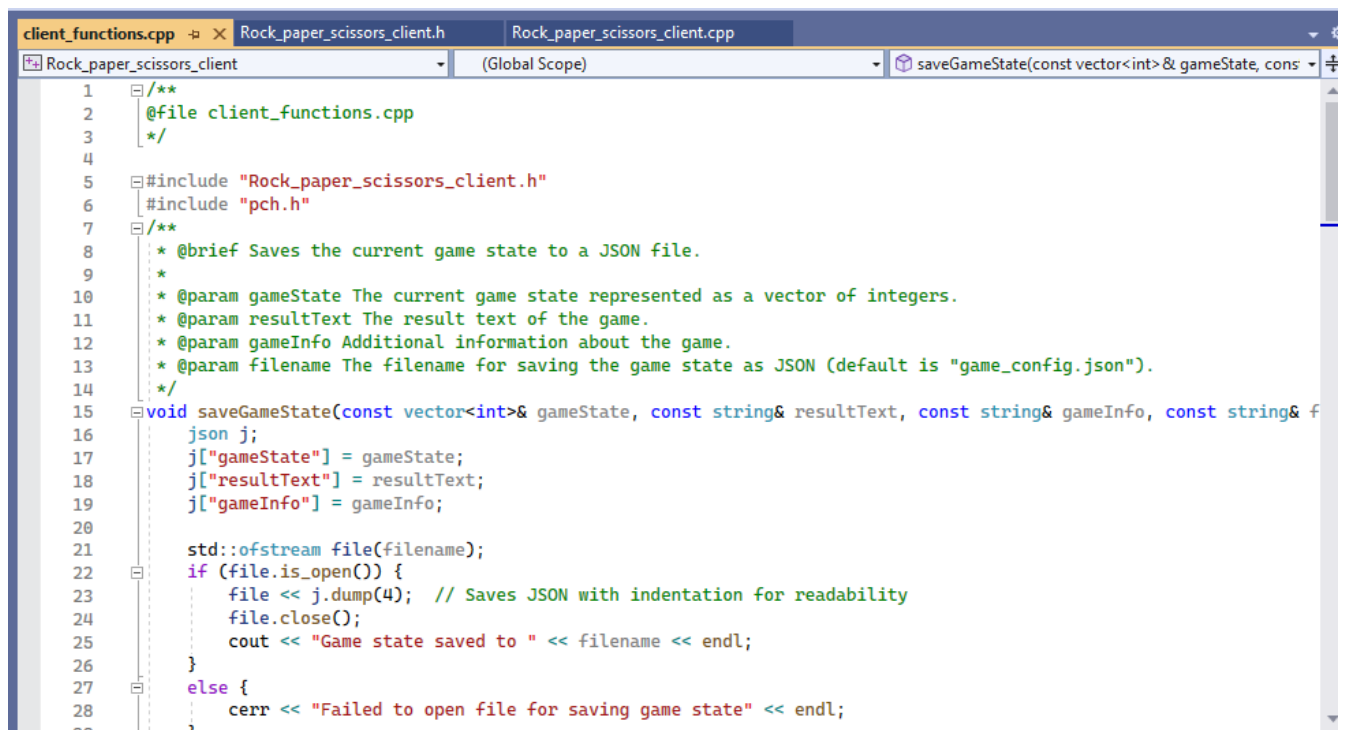
Хід роботи

1. Створив файл `Rock_paper_scissors_client.h`, куди переніс бібліотеки що використовуються та написав прототипи функцій.



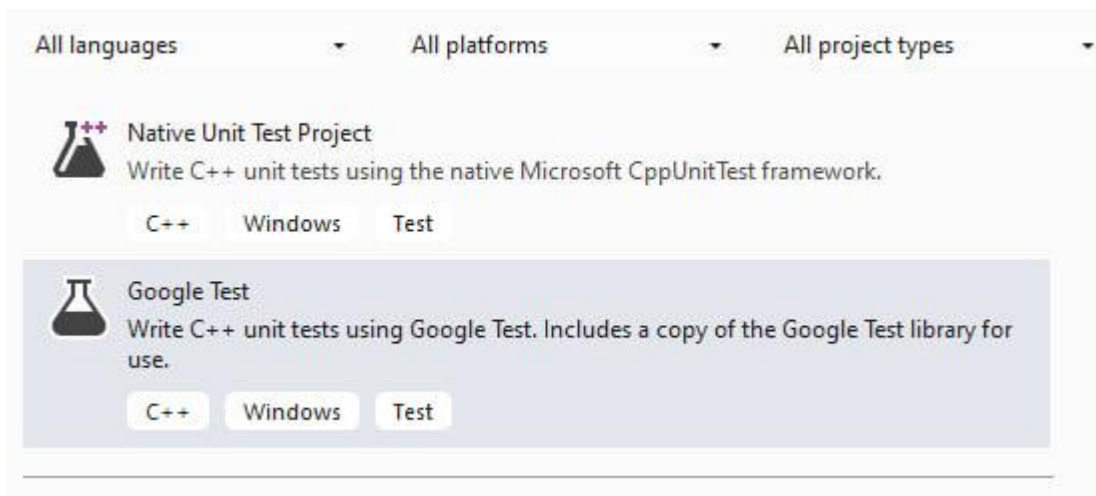
```
1  #pragma once
2  #include <iostream>
3  #include <windows.h>
4  #include <string>
5  #include <vector>
6  #include <cstdlib> // For system()
7  #include <fstream>
8  #include "nlohmann/json.hpp"
9
10 using json = nlohmann::json;
11 using namespace std;
12
13 bool loadGameState(vector<int>& gameState, string& resultText, string& gameInfo, const string& filename = "game_co
14 string createJSON(const vector<int>& array);
15 bool parseJSON(const string& json, vector<int>& array, string& resultText, string& gameInfo);
16 void saveGameState(const vector<int>& gameState, const string& resultText, const string& gameInfo, const string& f
```

2. Створив файл `client_functions.cpp`, куди переніс функції, що викликаються

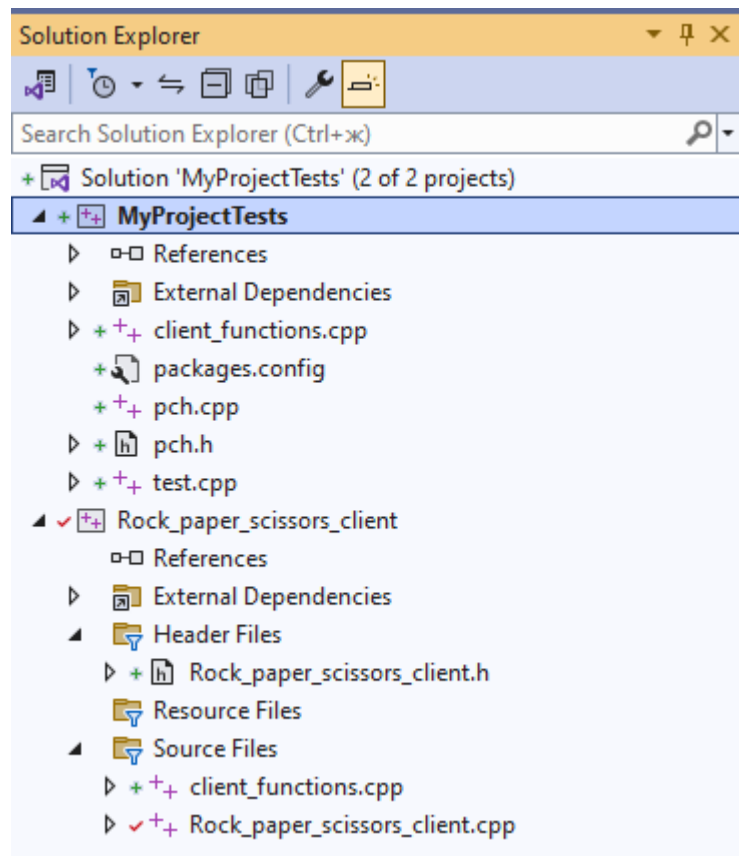


```
1  /**
2   * @file client_functions.cpp
3   */
4
5  #include "Rock_paper_scissors_client.h"
6  #include "pch.h"
7
8  /**
9   * @brief Saves the current game state to a JSON file.
10   *
11   * @param gameState The current game state represented as a vector of integers.
12   * @param resultText The result text of the game.
13   * @param gameInfo Additional information about the game.
14   * @param filename The filename for saving the game state as JSON (default is "game_config.json").
15   */
16 void saveGameState(const vector<int>& gameState, const string& resultText, const string& gameInfo, const string& f
17     json j;
18     j["gameState"] = gameState;
19     j["resultText"] = resultText;
20     j["gameInfo"] = gameInfo;
21
22     std::ofstream file(filename);
23     if (file.is_open()) {
24         file << j.dump(4); // Saves JSON with indentation for readability
25         file.close();
26         cout << "Game state saved to " << filename << endl;
27     }
28     else {
29         cerr << "Failed to open file for saving game state" << endl;
30     }
```

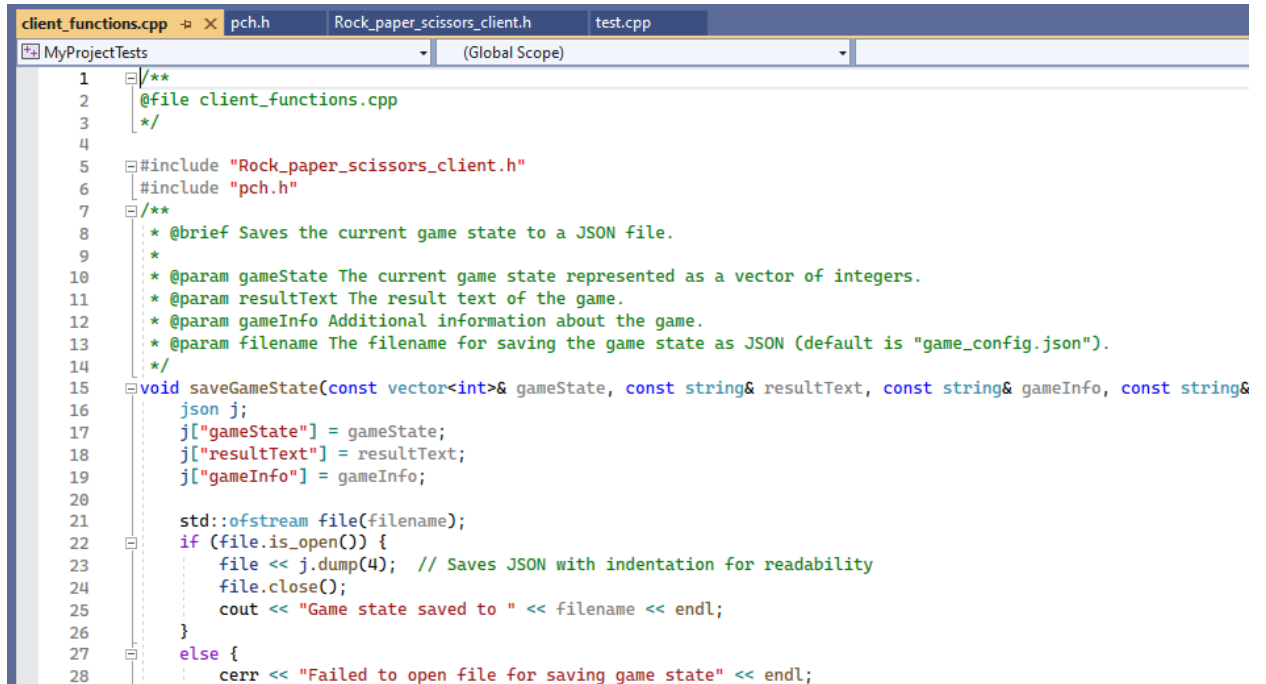
3. Створив новий проект для виконання тестів



4. Створив новий проект додав до нього існуючий проект:
Rock_paper_scissors_client

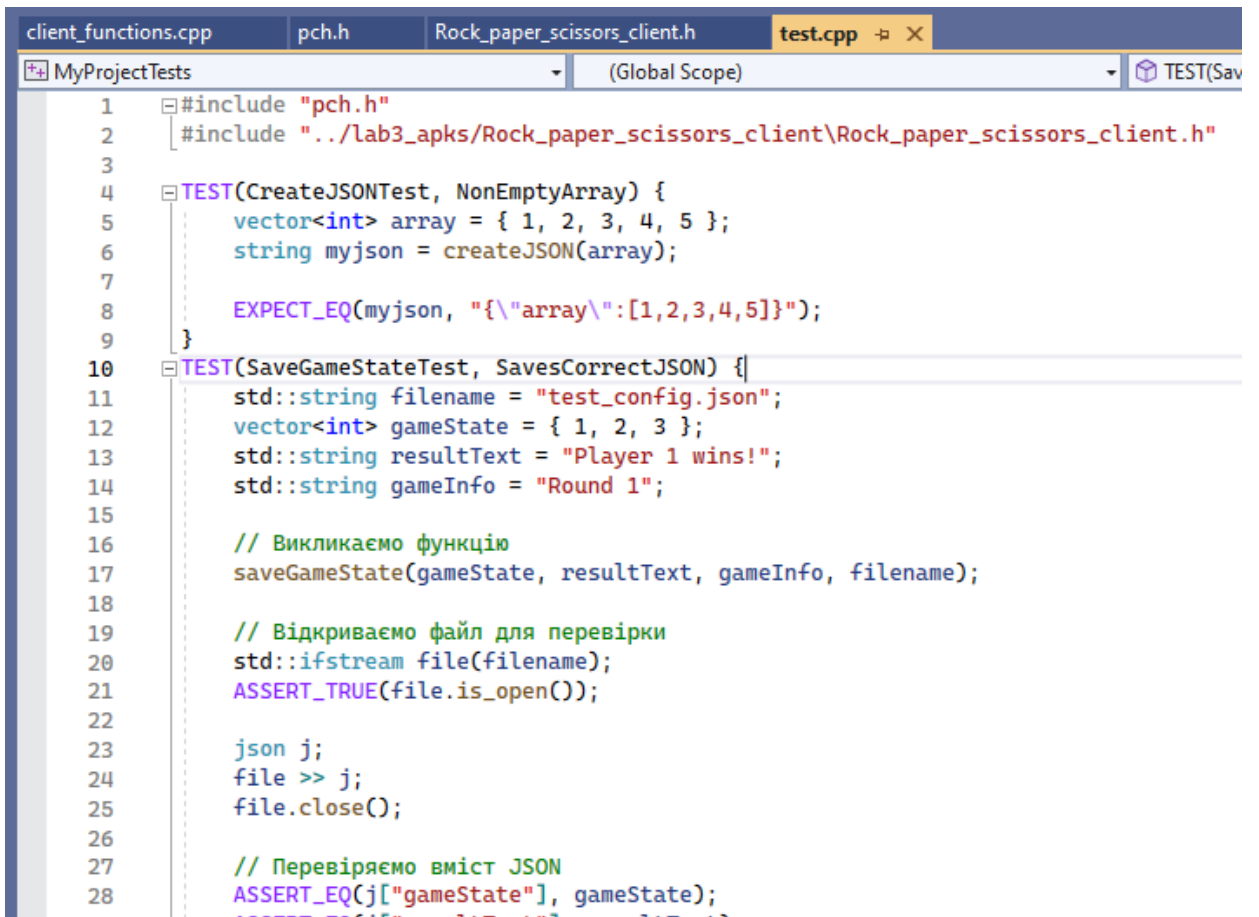


5. Додав до проекту файл з функціями



```
client_functions.cpp  pch.h  Rock_paper_scissors_client.h  test.cpp
MyProjectTests (Global Scope)
1  /**
2   * @file client_functions.cpp
3   */
4
5  #include "Rock_paper_scissors_client.h"
6  #include "pch.h"
7  /**
8   * @brief Saves the current game state to a JSON file.
9   *
10   * @param gameState The current game state represented as a vector of integers.
11   * @param resultText The result text of the game.
12   * @param gameInfo Additional information about the game.
13   * @param filename The filename for saving the game state as JSON (default is "game_config.json").
14   */
15  void saveGameState(const vector<int>& gameState, const string& resultText, const string& gameInfo, const string&
16      json j;
17      j["gameState"] = gameState;
18      j["resultText"] = resultText;
19      j["gameInfo"] = gameInfo;
20
21      std::ofstream file(filename);
22      if (file.is_open()) {
23          file << j.dump(4); // Saves JSON with indentation for readability
24          file.close();
25          cout << "Game state saved to " << filename << endl;
26      }
27      else {
28          cerr << "Failed to open file for saving game state" << endl;
```

6. Написав тести:



```
client_functions.cpp  pch.h  Rock_paper_scissors_client.h  test.cpp
MyProjectTests (Global Scope) TEST(Sav
1  #include "pch.h"
2  #include "../lab3_apks/Rock_paper_scissors_client/Rock_paper_scissors_client.h"
3
4  TEST(CreateJSONTest, NonEmptyArray) {
5      vector<int> array = { 1, 2, 3, 4, 5 };
6      string myjson = createJSON(array);
7
8      EXPECT_EQ(myjson, "{\"array\":[1,2,3,4,5]}");
9  }
10 TEST(SaveGameStateTest, SavesCorrectJSON) {
11     string filename = "test_config.json";
12     vector<int> gameState = { 1, 2, 3 };
13     std::string resultText = "Player 1 wins!";
14     std::string gameInfo = "Round 1";
15
16     // Викликаємо функцію
17     saveGameState(gameState, resultText, gameInfo, filename);
18
19     // Відкриваємо файл для перевірки
20     std::ifstream file(filename);
21     ASSERT_TRUE(file.is_open());
22
23     json j;
24     file >> j;
25     file.close();
26
27     // Перевіряємо вміст JSON
28     ASSERT_EQ(j["gameState"], gameState);
29     ASSERT_EQ(j["resultText"], resultText);
```

7. Усі тести виконались успішно

```
Microsoft Visual Studio Debug Console
Running main() from D:\a\_work\1\s\ThirdParty\googletest\googletest\src\gtest_main.cc
[=====] Running 4 tests from 4 test cases.
[-----] Global test environment set-up.
[-----] 1 test from CreateJSONTest
[ RUN ] CreateJSONTest.NonEmptyArray
[ OK ] CreateJSONTest.NonEmptyArray (0 ms)
[-----] 1 test from CreateJSONTest (1 ms total)

[-----] 1 test from SaveGameStateTest
[ RUN ] SaveGameStateTest.SavesCorrectJSON
Game state saved to test_config.json
[ OK ] SaveGameStateTest.SavesCorrectJSON (1 ms)
[-----] 1 test from SaveGameStateTest (1 ms total)

[-----] 1 test from LoadGameStateTest
[ RUN ] LoadGameStateTest.LoadsCorrectData
Game state loaded from test_config.json
[ OK ] LoadGameStateTest.LoadsCorrectData (1 ms)
[-----] 1 test from LoadGameStateTest (1 ms total)

[-----] 1 test from ParseJSONTest
[ RUN ] ParseJSONTest.ParsesCorrectData
[ OK ] ParseJSONTest.ParsesCorrectData (0 ms)
[-----] 1 test from ParseJSONTest (0 ms total)

[-----] Global test environment tear-down
[=====] 4 tests from 4 test cases ran. (6 ms total)
[ PASSED ] 4 tests.

D:\Downloads\github\csad2425ki410kovalchukao08\source\MyProjectTests\x64\Debug\MyProjectTests.exe (process 8836) exited with code 0.
Press any key to close this window . . .
```

8. Створив нову гілку feature/develop/task5. Створив Pull request для підтвердження змін в гілці develop, і надіслав запит на злиття викладачу.

Feature/develop/task5 #5

Open

Kovalchuk69 wants to merge 7 commits into `develop` from `feature/develop/task5`

Conversation 0

Commits 7

Checks 3

Files changed 78

+98,076 -132

Kovalchuk69 commented 2 minutes ago

No description provided.

Kovalchuk69 added 7 commits 1 hour ago

- ignore
- task5
- task5
- Create ci-tests.yml
- Update ci-tests.yml
- Update ci-tests.yml
- Update ci-tests.yml

Kovalchuk69 requested a review from anton-o-kitsera 2 minutes ago

Owner

Reviewers

Assignees

Labels

Projects

Milestone

Development

Notifications

Review requested

Review has been requested on this pull request. It is not required to merge. [Learn more about requesting a pull request review.](#)

Висновок:

У ході виконання лабораторної роботи розроблено тести для клієнтської частини програмного коду. Усі тести пройшли успішно.