# Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи №3 з дисципліни

«Основи програмування-2.

Модульне програмування»

«Класи та об’єкти»

Варіант 29

Виконав студент ІП-11 Тарасьонок Дмитро Євгенович

(шифр, прізвище, ім'я, по батькові)

Перевірив Вітковська Ірина Іванівна

( прізвище, ім'я, по батькові)

Київ 2022

Мета: Вивчити механізми створення і використання класів та об’єктів.

Умова задачі: Розробити клас, який є абстракцією тексту та підтримує операції додавання рядка до тексту і визначення частоти появи у тексті заданого символу кількість символів/загальна кількість символів). Створити масив об'єктів даного класу. Доповнити кілька текстів новими рядками. Визначити символ, який найчастіше зустрічається у текстах

Постановка задачі: для виконання поставленої задачі треба спроектувати клас, який матиме одне поле типу рядок. Для визначення частоти появи символів треба проходитись по всіх символах у рядку й додавати до лічильника 1 щоразу, як символ рядка дорівнює тому символу, який шукає користувач. Для визначення найпопулярнішого символу треба спочатку зберегти список усіх унікальних символів і підрахувати кількість кожного з цих символів у текстах.

Програма мовою C++:

Файл main.py

/\*

\* 1 курс, 2 семестр, Основи програмування, лабораторна робота №3.

\*

\* Варіант 29

\* Розробити клас, який є абстракцією тексту та підтримує операції додавання

\* рядка до тексту і визначення частоти появи у тексті заданого символу

\* (кількість символів/загальна кількість символів). Створити масив

\* об'єктів даного класу. Доповнити кілька текстів новими рядками.

\* Визначити символ, який найчастіше зустрічається у текстах

\*/

#include "Functions.h"

#include <locale>

#include <Windows.h>

using std::locale;

using std::cin;

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

locale::global(locale("rus\_rus.1251"));

String\* rgStrings = nullptr;

bool bBreak = false;

size\_t cCount = 0;

cout << "Вводьте рядки (Ctrl + G для завершення):" << endl;

while (!bBreak || cCount == 0)

{

string sLine;

char ch;

getline(cin, sLine);

int nBreakPos = sLine.find(0x7);

String sLineString = String(sLine.substr(0, nBreakPos));

if (sLineString.symbolsCount() != 0)

{

rgStrings = addToArray(rgStrings, sLineString, cCount);

}

bBreak = nBreakPos != -1;

}

cout << endl << "Список рядків:" << endl;

outputArray(rgStrings, cCount);

int iEditNum;

cout << "Введіть номер рядка для додавання тексту (0 для продовження): ";

cin >> iEditNum;

while (iEditNum != 0)

{

if (iEditNum < 0 || cCount <= iEditNum)

{

cout << "Введено невірний номер!" << endl;

}

string sLine;

cout << "Введіть рядок, яким хочете доповнити обраний: ";

cin.ignore();

getline(cin, sLine);

rgStrings[iEditNum - 1].addText(sLine);

cout << "Введіть номер рядка для додавання тексту (0 для продовження): ";

cin >> iEditNum;

}

cout << endl << "Список рядків:" << endl;

outputArray(rgStrings, cCount);

size\_t nRepeats = 0;

char chMostPopular = getMostPopularSymbol(rgStrings, cCount, nRepeats);

cout << "Найчастіше зустрічається символ " << '\'' << chMostPopular << '\'' << endl

<< "Кількість цього символу в усіх рядка: " << nRepeats << endl;

}

Файл String.h

#pragma once

#include <string>

#include <iostream>

using std::string;

using std::ostream;

using std::cout;

class String

{

public:

String(const string& a\_s="");

size\_t symbolsCount() const { return m\_sStr.length(); }

String& addText(const String& a\_s);

String& addText(const string& a\_s);

String& setText(const String& a\_s);

String& setText(const string& a\_s);

string getString() const { return m\_sStr; }

char getSymbol(size\_t pos) const { return m\_sStr[pos]; }

size\_t count(char c) const;

void output(ostream& os=cout) { os << m\_sStr; }

private:

string m\_sStr;

};

Файл String.cpp

#include "String.h"

String::String(const string& a\_s)

:m\_sStr{ a\_s }

{}

String& String::addText(const String& a\_s)

{

m\_sStr += a\_s.getString();

return \*this;

}

String& String::addText(const string& a\_s)

{

m\_sStr += a\_s;

return \*this;

}

String& String::setText(const String& a\_s)

{

m\_sStr = a\_s.getString();

return \*this;

}

String& String::setText(const string& a\_s)

{

m\_sStr = a\_s;

return \*this;

}

size\_t String::count(char c) const

{

size\_t cCounter = 0;

for (size\_t i = 0; i < symbolsCount(); ++i)

{

if (m\_sStr[i] == c)

{

cCounter += 1;

}

}

return cCounter;

}

Файл Functions.h

#pragma once

#include "String.h"

using std::endl;

String\* addToArray(String\* a\_rgOldArr, const String& a\_sNew, size\_t& a\_nCurSize);

char\* addToArray(char\* a\_rgOldArr, char a\_chNew, size\_t& a\_nCurSize);

void outputArray(String\* a\_rgArr, size\_t a\_nSize);

bool isInArr(char\* a\_rgArr, size\_t a\_nSize, char a\_ch);

char getMostPopularSymbol(const String\* a\_rgArr, size\_t a\_nArrSize, size\_t& a\_nRepeats);

Файл Functions.cpp

#include "Functions.h"

String\* addToArray(String\* a\_rgOldArr, const String& a\_sNew, size\_t& a\_nCurSize)

{

String\* newArr = new String[a\_nCurSize + 1];

for (size\_t i = 0; i < a\_nCurSize; ++i)

{

newArr[i] = a\_rgOldArr[i];

}

if (a\_rgOldArr != nullptr)

{

delete[] a\_rgOldArr;

}

newArr[a\_nCurSize] = a\_sNew;

++a\_nCurSize;

return newArr;

}

char\* addToArray(char\* a\_rgOldArr, char a\_chNew, size\_t& a\_nCurSize)

{

char\* newArr = new char[a\_nCurSize + 1];

for (size\_t i = 0; i < a\_nCurSize; ++i)

{

newArr[i] = a\_rgOldArr[i];

}

if (a\_rgOldArr != nullptr)

{

delete[] a\_rgOldArr;

}

newArr[a\_nCurSize] = a\_chNew;

++a\_nCurSize;

return newArr;

}

void outputArray(String\* a\_rgArr, size\_t a\_nSize)

{

for (size\_t i = 0; i < a\_nSize; ++i)

{

cout << i + 1 << ". ";

a\_rgArr[i].output(cout);

cout << endl;

}

}

bool isInArr(char\* a\_rgArr, size\_t a\_nSize, char a\_ch)

{

bool bIsInArr = false;

for (size\_t i = 0; i < a\_nSize; ++i)

{

if (a\_rgArr[i] == a\_ch)

{

bIsInArr = true;

}

}

return bIsInArr;

}

char getMostPopularSymbol(const String\* a\_rgArr, size\_t a\_nArrSize, size\_t& a\_nRepeats)

{

char\* rgUniqueChars = nullptr;

size\_t nUniqueSize = 0;

for (size\_t i = 0; i < a\_nArrSize; ++i)

{

for (size\_t j = 0; j < a\_rgArr[i].symbolsCount(); ++j)

{

if (!isInArr(rgUniqueChars, nUniqueSize, a\_rgArr[i].getSymbol(j)))

{

rgUniqueChars = addToArray(rgUniqueChars, a\_rgArr[i].getSymbol(j), nUniqueSize);

}

}

}

char chMostPopular = a\_rgArr[0].getSymbol(0);

a\_nRepeats = 0;

for (size\_t i = 0; i < nUniqueSize; ++i)

{

char chToFind = rgUniqueChars[i];

size\_t nLocalRepeats = 0;

for (int j = 0; j < a\_nArrSize; ++j)

{

nLocalRepeats += a\_rgArr[j].count(chToFind);

}

if (a\_nRepeats < nLocalRepeats)

{

chMostPopular = chToFind;

a\_nRepeats = nLocalRepeats;

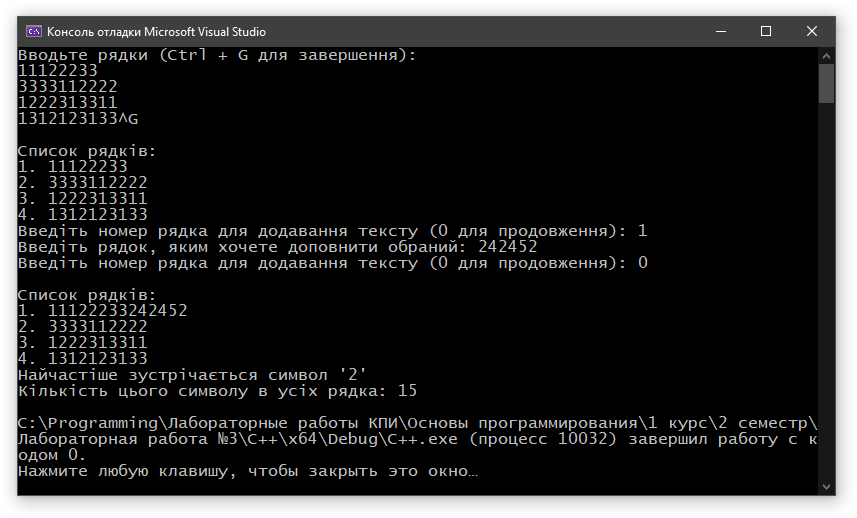
}

}

return chMostPopular;

}

Виконання коду мовою C++:



Висновок: під час виконання даної лабораторної роботи я здобув практичних навичок у створенні класів. Дослідив модифікатори доступу до членів класу (private, protected, public), вивчив використання геттерів та сеттерів на прикладу зміни та отримання тексту в спроектованому класі.