

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Механіко-технологічний факультет

ЗВІТ
ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 10
з навчальної дисципліни
“Базові методології та технології програмування”
РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ ОБРОБЛЕННЯ
ДАННИХ СКВОДКИХ ТИПІВ
З ФАЙЛАМИ ВВЕДЕННЯ/ВИВЕДЕННЯ

ВИКОНАВ
студент академічної групи КН-22
Кічура М.Г.

ПЕРЕВІРИВ
ст. викладач кафедри кібербезпеки
та програмного забезпечення
Олександр Собінов

Лабораторна робота №10

Мета роботи: полягає у набутті ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою C++ програмних модулів створення й оброблення даних типів масив, структура, об'єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних бібліотек для оброблення символічної інформації.

<https://github.com/odorenskyi/>

ВАРІАНТ 20

— *ВХІДНИЙ ТЕКСТ - ВМІСТ ВХІДНОГО ТЕКСТОВОГО ФАЙЛУ* —

Довільне речення українською мовою.

— *ЗАДАЧА 10.1* —

У вихідний текстовий файл записати:

- авторську інформацію: ім'я й прізвище розробника модуля, установа/організація, місто, країна, рік розробки;
- кількість знаків «:» у реченні із вхідного файлу;
- речення із вхідного файлу, у якому відсутні всі голосні літери.

— *ЗАДАЧА 10.2* —

У вхідний текстовий файл дописати:

- транслітерований текст з цього файлу відповідно до Постанови Кабінету Міністрів України “Про впорядкування транслітерації українського алфавіту латиницею” від 27 січня 2010 року № 55;
- дату дозапису інформації у форматі «ДД / ММ / РР».

— *ЗАДАЧА 10.3* —

Вхідні дані – числові значення x , y , z та натуральне число b . У вихідний текстовий файл дописати:

- результати виконання функцій із заголовкового файлу `Modules/Прізвище.h` `s_calculation` з аргументами x , y , z ;
- число b у двійковому коді.



- Мова повідомлень – українська (наприклад, якщо у вихідний файл записується кількість символів у вхідному файлі, то модуль повинен сформулювати й записати/дописати повноцінне речення: “У файлі `ВхФайл.txt` міститься 257 символів.”).
- Вхідний файл `*.txt` створиться користувачем, у який за допомогою текстового редактора (у Windows – Блокнот) записується вхідний текст відповідно до завдання; вихідний файл створюється програмним модулем; імена вхідного й вихідного файлів є параметрами відповідного модуля.
- Перед читання/записом з/у файловий потік слід реалізувати перевірку його відкриття; після завершення – закрити всі відкриті файлові потоки.
- Оброблення текстових файлів рекомендовано реалізувати за допомогою файлових потоків `ofstream` та `ifstream` `<fstream> C++`.
- Для отримання локальної дати й часу ОС можна використати стандартні функції `time`, `ctime`, `localtime`, `asctime`, реалізовані у `ctime / time.h`.

Лістинг Коду з ModulesMax

```

#include "Modules.h"
#include <math.h>
#include <fstream>
#include <iostream>
#include <string>
#include <algorithm>
#include <locale>
#include <time.h>
#include <map>
#include <unordered_map>
#include <bitset>
using namespace std;
double s_calculation(double x,double y,double z)
{ double S=(1/2)*pow(x,2)-sqrt(fabs(pow((y+z),2)-pow(x,5)))-log(fabs(sin(z)));
  return S;

}

std::string task10_1(const string& filename)
{
  setlocale(LC_ALL, "uk_UA.UTF-8");
  ofstream file(filename, ios::app);
  if(file.is_open())
  {
    file << "Осадчий В.К" << endl
    << " ЦНТУ" << endl
    << " Кропивницький" << endl
    << " Україна" << endl
    << " 2023" << endl << endl;
    cout << "У файл була успішно записана інформація" << endl;
    file.close();
  }
  else
  {
    cout << "Помилка!Такого файлу немає" << endl;
  }
}

int task10_1_2(const string& filename)
{ setlocale(LC_ALL, "uk_UA.UTF-8");
  ifstream file(filename);
  if (file.is_open())
  {
    int count = 0;
    string line;
    while (getline(file, line))
    {
      count +=std::count(line.begin(), line.end(), ':');
    }
  }
}

```

```

    cout<<count<<endl;
    return count;
}
}
else
{
    cout<<"Помилка!Такого файлу немає"<< endl;
    return 0;
}
}
int task10_1_3(const std::string& filename)
{
    setlocale(LC_ALL, "ukr");
    ifstream file(filename);
    if (!file.is_open()) {
        cerr << "Помилка! Неможливо відкрити файл " << filename << endl;
        return -1;
    }
    string line;
    string sentence;
    while (getline(file, line)) {
        for (char c : line) {
            if (c == '.' || c == '!' || c == '?') {
                // End of sentence
                string sentence_without_vowels;
                for (char s : sentence) {
                    if (s != 'a' && s != 'e' && s != 'i' && s != 'o' && s != 'u' &&
                        s != 'A' && s != 'E' && s != 'I' && s != 'O' && s != 'U') {
                        sentence_without_vowels += s;
                    }
                }
                cout << sentence_without_vowels << c << " ";
                sentence.clear();
            } else {
                sentence += c;
            }
        }
    }
    file.close();
    return 0;
}

std::string transliterate_text(const string& input) {
    setlocale(LC_ALL, "uk-UA.UTF-8");
    string output = "";
    unordered_map<char, string> translit_map {
        {'À', "A"}, {'Á', "B"}, {'Â', "V"}, {'Ã', "H"}, {'Ÿ', "G"},
        {'Ä', "D"}, {'Å', "E"}, {'à', "Ye"}, {'Æ', "Zh"}, {'Ç', "Z"},
        {'È', "Y"}, {'²', "I"}, {'ı', "Yi"}, {'É', "Y"}, {'Ê', "K"},

```

```

    {'Ë', "L"}, {'Ì', "M"}, {'Í', "N"}, {'Î', "O"}, {'Ï', "P"},
    {'Ð', "R"}, {'Ñ', "S"}, {'Ò', "T"}, {'Ó', "U"}, {'Ô', "F"},
    {'Õ', "Kh"}, {'Ö', "Ts"}, {'×', "Ch"}, {'Ø', "Sh"}, {'Ù', "Shch"},
    {'Ü', ""}, {'Ð', "Yu"}, {'ß', "Ya"}, {'à', "a"}, {'á', "b"},
    {'â', "v"}, {'ã', "h"}, {'ä', "g"}, {'ä', "d"}, {'å', "e"},
    {'ø', "ie"}, {'æ', "zh"}, {'ç', "z"}, {'è', "y"}, {'ì', "i"},
    {'í', "i"}, {'é', "i"}, {'ê', "k"}, {'ë', "l"}, {'î', "m"},
    {'í', "n"}, {'î', "o"}, {'ï', "p"}, {'ð', "r"}, {'ñ', "s"},
    {'ò', "t"}, {'ó', "u"}, {'ô', "f"}, {'õ', "kh"}, {'ö', "ts"},
    {'÷', "ch"}, {'ø', "sh"}, {'ù', "shch"}, {'ü', ""}, {'þ', "iu"},
    {'ÿ', "ia"}
};

for (const char& c : input) {
    if (translit_map.count(c) > 0) {
        output += translit_map[c];
        cout<<output<<endl;
    }
    else {
        output += c;
    }
}
return output;
}

void task10_2(const string& filename) {
    setlocale(LC_ALL, "uk_UA.UTF-8");
    time_t now = time(0);
    tm* ltm = localtime(&now);
    string date = to_string(ltm->tm_mday) + "\\" + to_string(ltm->tm_mon + 1) + "\\" +
to_string(ltm->tm_year + 1900);
    ifstream input_file(filename);
    ofstream output_file("output.txt");
    if (input_file.is_open() && output_file.is_open()) {
        string line;
        while (getline(input_file, line)) {
            string transliterated_line = transliterate_text(line);
            output_file << transliterated_line << endl;
        }
        output_file << date << endl;
        input_file.close();
        output_file.close();

        cout << "Текст успішно доданий до файлу з назвою 'output.txt'." <<endl;
    }
    else {
        cout << "Помилка!Такого файлу немає" <<endl;
    }
}

```

```
double task10_3(double x, double y, double z, int b) {
    setlocale(LC_ALL, "uk_UA.UTF-8");
    ofstream outfile("output.txt", ios_base::app);
    double result = s_calculation(x, y, z);
    outfile << " Результат: " << result << endl;
    outfile << "b у двійковій системі числення: " << bitset<32>(b) << endl;
    outfile.close();
    cout << "Дані успішно додані до файлу 'output.txt'." << endl;
}
```

Лістинг з Modules.h

```
#ifndef MAIN_H_INCLUDED
#define MAIN_H_INCLUDED
#include <string>
double s_calculation(double x,double y,double z);
std::string task10_1(const std::string& filename);
int task10_1_2(const std::string& filename);
int task10_1_3(const std::string& filename);
void task10_2(const std::string& filename);
std::string transliterate_text(const std::string& text);
double task10_3(double x, double y, double z, int b);
#endif // MAIN_H_INCLUDED
```

Лістинг коду з TestDriver

```
#include <iostream>
#include <Modules.h>
#include <string>
using namespace std;

int main()
{ std::string file,file2,file3,file4;
  cin>>file;
  cin>>file2;
  cin>>file3;
  cin>>file4;
  task10_1(file);
  task10_1_2(file2);
  task10_1_3(file3);
  task10_2(file4);
  double x,y,z;
  int b;
  cin>>x;
  cin>>y;
  cin>>z;
  cin>>b;
  cout<<task10_3(x,y,z,b)<<endl;
```

```
    return 0;  
}
```

Висновок: Мета цієї роботи полягала у набутті практичних умінь і навичок застосування теоретичних положень методології модульного програмування реалізації метода роботи з файлами виведення/виведення мовою програмування C++ з використанням кросплатформенного середовища Codeblocks

У данній лабораторній роботі ми зосередилися на створення й оброблення даних типів: масив, структура, об'єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних бібліотек для оброблення символьної інформації після завершення роботи я отримав достатньо знань та навичок щоб виконувати схожі завдання.