

Міністерство освіти і науки, молоді та спорту України
Національний університет “Львівська політехніка”

Кафедра ЕОМ



Звіт

з домашнє завдання №14 з дисципліни:

“Алгоритми та моделі обчислень”

Варіант: № 24.

Виконав:

ст. групи КІ-203

Ширий Богдан Ігорович

Перевірів:

ст. викладач кафедри ЕОМ

Козак Назар Богданович

ЗАВДАННЯ:

УМОВА:

Написати на C/C++ програму, яка реалізує алгоритм Бояра-Мура для знаходження у вхідній стрічці всіх входжень заданого слова.

ВИБІР ВАРІАНТУ:

Шукане слово це моє прізвище латинськими літерами: Shyryi.

ВИКОНАННЯ:

Склав C++ десктопну програму, виконання якої зобразив на рисунках 1 та 2.

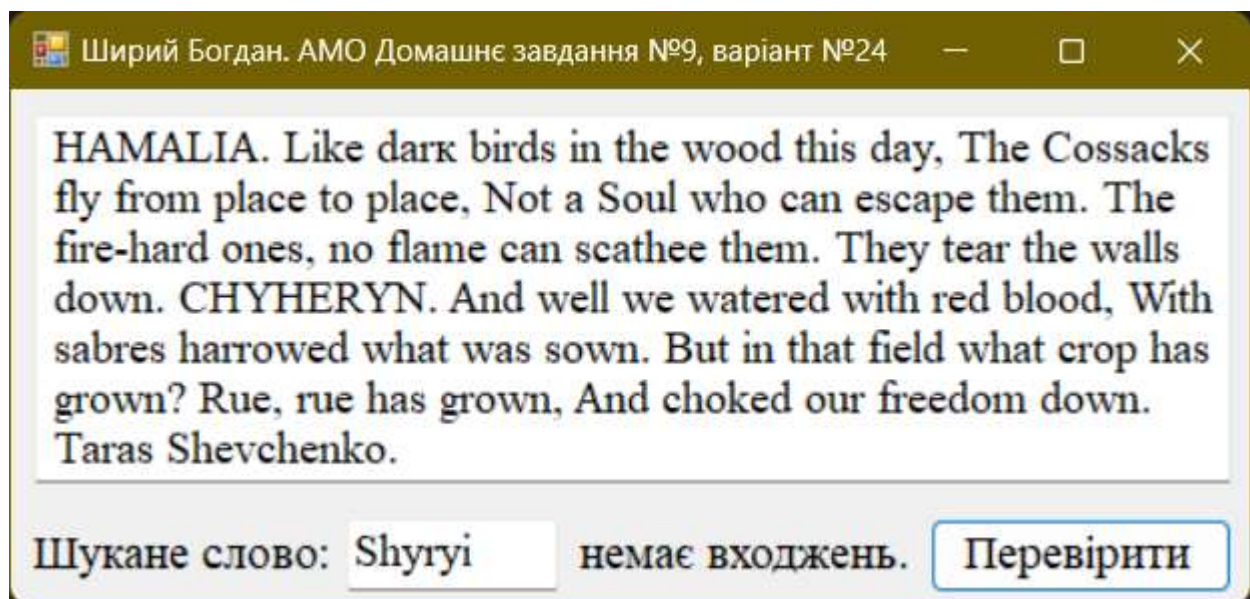


Рис. 1. Початкові дані програми без Shyryi.

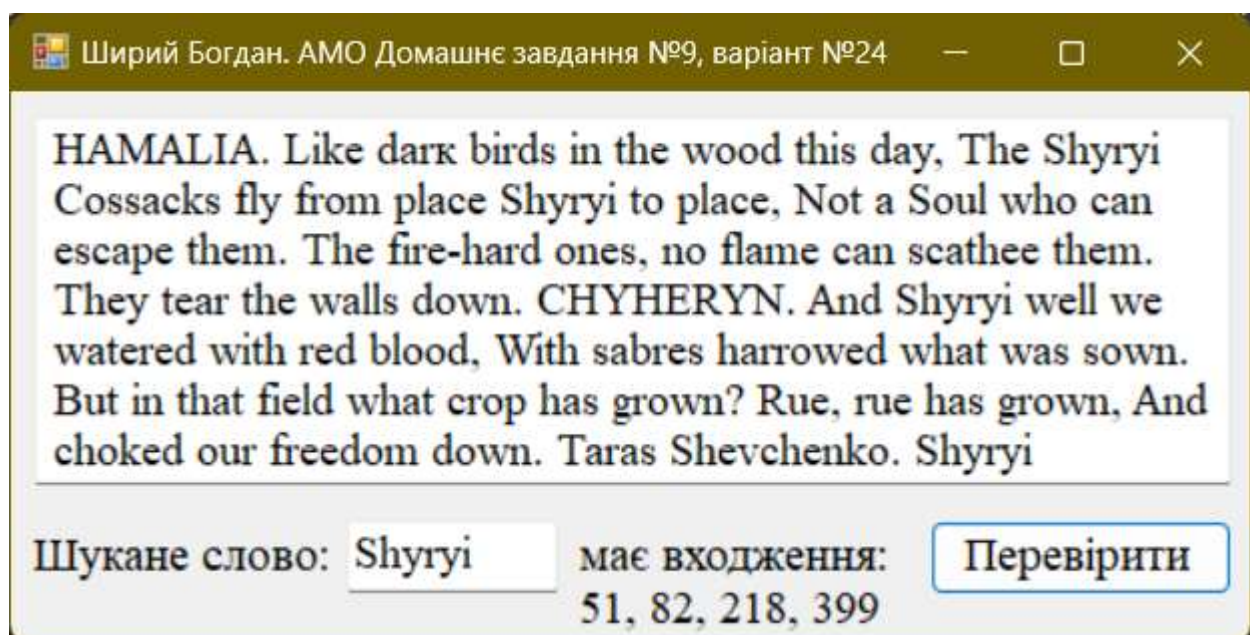


Рис. 2. Початкові дані програми з Shyryi.

Код кожного файлу десктопної програми навів у лістингах:

- MainForm.cpp у лістигу 1,
- MainForm.h у лістингу 2.

Лістинг 1. Код файлу MainForm.cpp.

```
#include "MainForm.h"

using namespace System;
using namespace System::Windows::Forms;
[STAThreadAttribute]
int main() {
    Application::SetCompatibleTextRenderingDefault(false);
    Application::EnableVisualStyles();
    acmShyryiHW14::MainForm form;
    Application::Run(% form);
}
```

Лістинг 2. Код заголовкового файлу MainForm.h.

```
#pragma once

#include <iostream>
#include <cstring>
#include <algorithm>
using namespace std;

#define MAX_CHAR 256

System::String^ boyer_moore_search(System::String^ text, System::String^ pattern) {
    int n = text->Length;
    int m = pattern->Length;

    int bad_char[MAX_CHAR];
    memset(bad_char, -1, sizeof bad_char);
    for (int i = 0; i < m; i++) {
        bad_char[(int)pattern[i]] = i;
    }

    System::String^ indexes = "";
    int shift = 0;
    while (shift <= n - m) {
        int j = m - 1;
        while (j >= 0 && pattern[j] == text[shift + j]) {
            j--;
        }
        if (j < 0) {
            if (indexes != "") {
                indexes += ", ";
            }
            indexes += shift;
            shift += (shift + m < n) ? m - bad_char[(int)text[shift + m]] : 1;
        }
        else {
            shift += max(1, j - bad_char[(int)text[shift + j]]);
        }
    }
    return indexes;
}

namespace acmShyryiHW14 {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;

    /// <summary>
    /// Summary for MainForm

```

```

/// </summary>
public ref class MainForm : public System::Windows::Forms::Form
{
public:
    MainForm(void)
    {
        InitializeComponent();
        //
        //TODO: Add the constructor code here
        //
    }

protected:
    /// <summary>
    /// Clean up any resources being used.
    /// </summary>
    ~MainForm()
    {
        if (components)
        {
            delete components;
        }
    }

private: System::Windows::Forms::TextBox^ inputTextTextBox;
protected:
private: System::Windows::Forms::Button^ checkButton;
private: System::Windows::Forms::TextBox^ inputPatternTextBox;
private: System::Windows::Forms::Label^ infoLabel;
private: System::Windows::Forms::Label^ outputLabel;

private:
    /// <summary>
    /// Required designer variable.
    /// </summary>
    System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    void InitializeComponent(void)
    {
        this->inputTextTextBox = (gcnew System::Windows::Forms::TextBox());
        this->checkButton = (gcnew System::Windows::Forms::Button());
        this->inputPatternTextBox = (gcnew System::Windows::Forms::TextBox());
        this->infoLabel = (gcnew System::Windows::Forms::Label());
        this->outputLabel = (gcnew System::Windows::Forms::Label());
        this->SuspendLayout();
        //
        // inputTextTextBox
        //
        this->inputTextTextBox->Font = (gcnew System::Drawing::Font(L"Times New
Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->inputTextTextBox->Location = System::Drawing::Point(12, 12);
        this->inputTextTextBox->Multiline = true;
        this->inputTextTextBox->Name = L"inputTextTextBox";
        this->inputTextTextBox->Size = System::Drawing::Size(637, 180);
        this->inputTextTextBox->TabIndex = 0;
        //
        // checkButton
        //
        this->checkButton->Font = (gcnew System::Drawing::Font(L"Times New Roman",
13.8F));
        this->checkButton->Location = System::Drawing::Point(489, 210);
        this->checkButton->Name = L"checkButton";
        this->checkButton->Size = System::Drawing::Size(161, 37);
        this->checkButton->TabIndex = 1;
        this->checkButton->Text = L"Перевірити";
        this->checkButton->UseVisualStyleBackColor = true;
        this->checkButton->Click += gcnew System::EventHandler(this,
&MainForm::checkButton_Click);
        //
        // inputPatternTextBox
        //
        this->inputPatternTextBox->Font = (gcnew System::Drawing::Font(L"Times New
Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->inputPatternTextBox->Location = System::Drawing::Point(179, 210);
        this->inputPatternTextBox->Name = L"inputPatternTextBox";
    }

```

```

        this->inputPatternTextBox->Size = System::Drawing::Size(112, 34);
        this->inputPatternTextBox->TabIndex = 2;
        //
        // infoLabel
        //
        this->infoLabel->AutoSize = true;
        this->infoLabel->Font = (gcnew System::Drawing::Font(L"Times New Roman",
13.8F));

        this->infoLabel->Location = System::Drawing::Point(7, 215);
        this->infoLabel->Name = L"infoLabel";
        this->infoLabel->Size = System::Drawing::Size(155, 26);
        this->infoLabel->TabIndex = 3;
        this->infoLabel->Text = L"Шукане слово:";
        //
        // outputLabel
        //
        this->outputLabel->AutoSize = true;
        this->outputLabel->Font = (gcnew System::Drawing::Font(L"Times New Roman",
13.8F));

        this->outputLabel->Location = System::Drawing::Point(297, 215);
        this->outputLabel->Name = L"outputLabel";
        this->outputLabel->Size = System::Drawing::Size(0, 26);
        this->outputLabel->TabIndex = 4;
        //
        // MainForm
        //
        this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);
        this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(662, 253);
        this->Controls->Add(this->outputLabel);
        this->Controls->Add(this->infoLabel);
        this->Controls->Add(this->inputPatternTextBox);
        this->Controls->Add(this->checkBox);
        this->Controls->Add(this->inputTextTextBox);
        this->Name = L"MainForm";
        this->Text = L"Ширій Богдан. АМО Домашнє завдання №9, варіант №24";
        this->ResumeLayout(false);
        this->PerformLayout();
    }
#pragma endregion
    private: System::Void checkBox_Click(System::Object^ sender, System::EventArgs^ e) {
        try
        {
            if (boyer_moore_search(inputTextTextBox->Text, inputPatternTextBox->Text) ==
"" ) {
                outputLabel->Text = "немає входжень.";
                this->ClientSize = System::Drawing::Size(495, 206);
            }
            else
            {
                outputLabel->Text = "має входження:\n" +
boyer_moore_search(inputTextTextBox->Text, inputPatternTextBox->Text);
                this->ClientSize = System::Drawing::Size(495, 220);
            }
        }
        catch (System::Exception^ e)
        {
            outputLabel->Text = "Виникла помилка.";
        }
    }
};
}

```