МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Казанский национальный исследовательский технический университет

им. А.Н. Туполева – КАИ»

Институт компьютерных технологий и защиты информации

Отделение СПО в ИКТЗИ (Колледж информационных технологий)

ЛАБОРАТОРНАЯ РАБОТА №3

по дисциплине

Теория Алгоритмов

Тема: «РАЗРАБОТКА АЛГОРИТМА ПОИСКА МАКСИМАЛЬНОГО ПОТОКА В ТРАНСПОРТНОЙ СЕТИ»

Работу выполнил

Студент гр.4338

Бусов В.Р.

Принял

Преподаватель Валова П.А.

Казань 2024

ЦЕЛЬ

Разработать алгоритм поиска максимального потока в транспортной сети.

ХОД РАБОТЫ

1. Самостоятельно задать пропускные способности дуг и построить максимальный поток в транспортной сети.

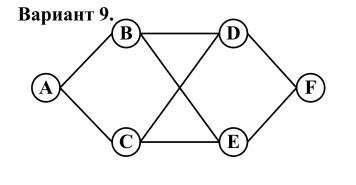


Рисунок 1 - Задание

|  |  |
| --- | --- |
| Данная сеть | Сеть с нулевым потоком |
| Увеличивающая цепь: AС, СD, DF; направление дуг совпадает с направлением потока, δ = min(2 – 0, 6 – 0, 7 – 0) = 2. | Новые потоки по дугам цепи: AС: 0+2=2, СD: 0 +2=2, DF: 0+2=2: |
| Увеличивающая цепь: AB, BD, DF; направление дуг совпадает с направлением потока, δ = min(1 – 0, 5 – 0, 7 – 2) = 1. | Новые потоки по дугамцепи: AB: 0+1=1, BD: 0+1=1, DF: 2+1=3. |

Увеличивающих цепей в сети нет, поэтому максимальный поток построен, и он равен 3 = 1 + 2 = 3 + 0.

2. Найти минимальный разрез сети и проверить справедливость теоремы Форда–Фалкерсона.

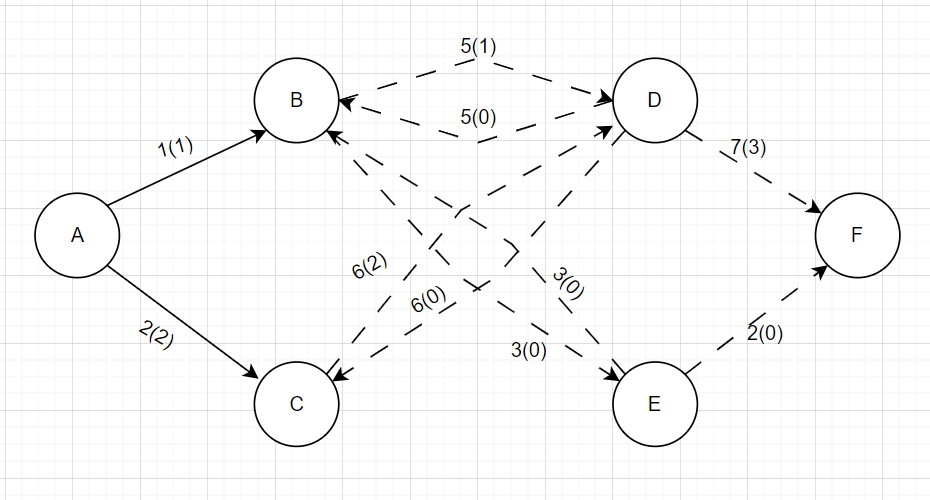


Рисунок 2 – Разрез 1

Пропускная способность: 1+2=3. Поток: 1+2=3

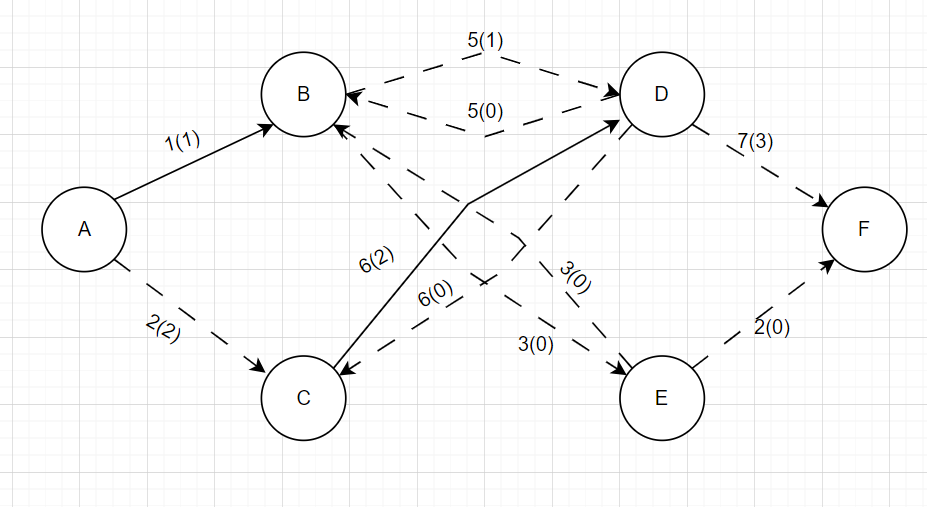


Рисунок 3 – Разрез 2

Пропускная способность: 1+6= 7. Поток: 1 + 2 = 3

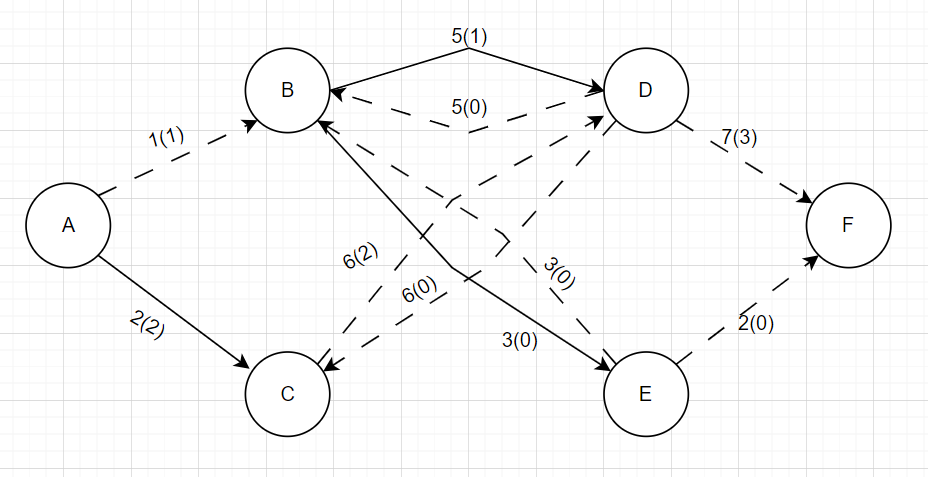


Рисунок 4 – Разрез 3

Пропускная способность: 2+5+3=10. Поток: 2+1+0=3

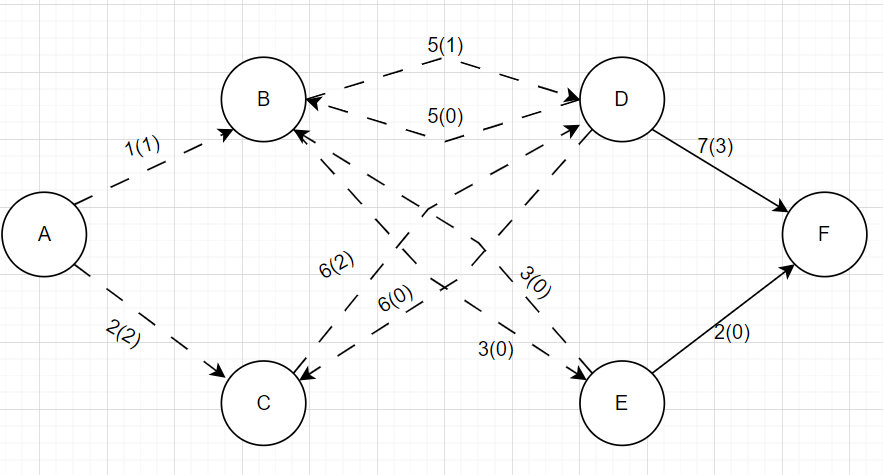


Рисунок 5 – Разрез 4

Пропускная способность: 7+2=9. Поток: 3+0=3

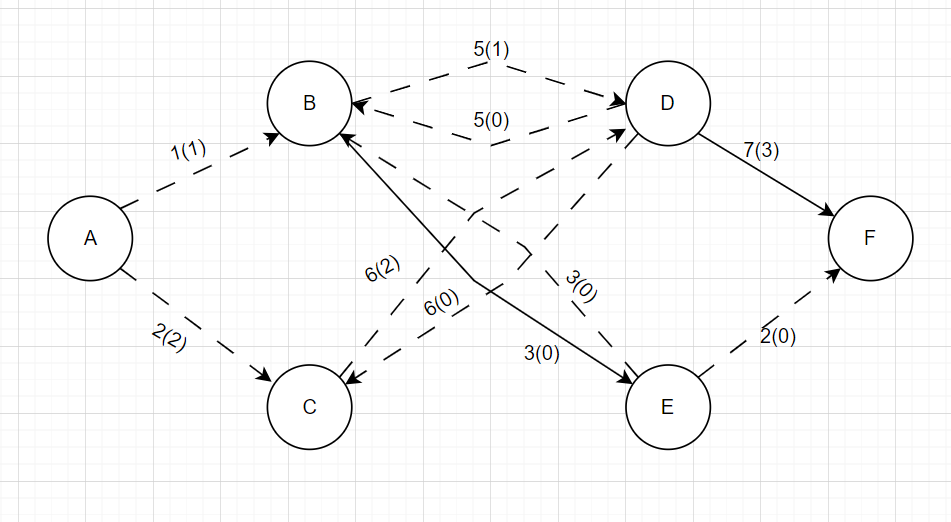


Рисунок 6 – Разрез 5

Пропускная способность: 7+3=10. Поток: 3+0=3

**Часть 2.**

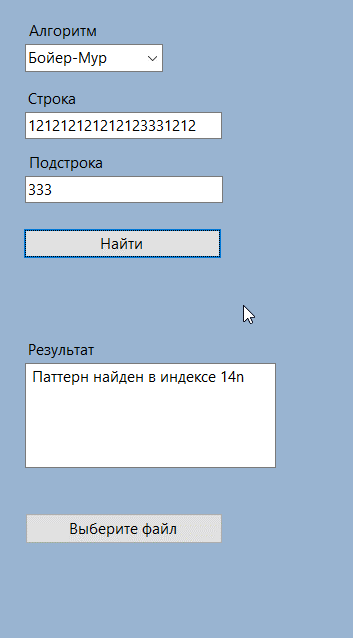


Рисунок 6 – Результат Бойера-Мура

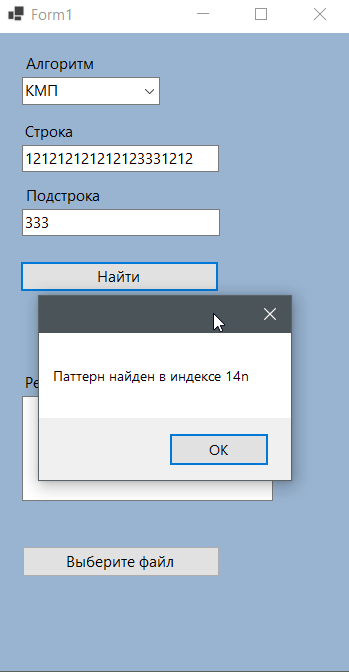


Рисунок 7 – Результат Бойер-Мур

**Листинг программы**

using System.Text;

namespace WinFormsApp3

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

algorithmComboBox.Items.Add("КМП");

algorithmComboBox.Items.Add("Бойер-Мур");

algorithmComboBox.SelectedIndex = 0;

}

private void searchButton\_Click(object sender, EventArgs e)

{

string text = textInput.Text;

string pattern = patternInput.Text;

if (string.IsNullOrEmpty(text) || string.IsNullOrEmpty(pattern))

{

MessageBox.Show("Пожалуйста, введите строку и подстроку.");

return;

}

string algorithm = algorithmComboBox.SelectedItem.ToString();

resultOutput.Clear();

if (algorithm == "КМП")

{

KMPSearch(text, pattern);

}

else if (algorithm == "Бойер-Мур")

{

BoyerMooreSearch(text, pattern);

}

}

private void openFileButton\_Click(object sender, EventArgs e)

{

using (OpenFileDialog openFileDialog = new OpenFileDialog())

{

openFileDialog.InitialDirectory = @"C:"; // Укажите начальную директорию, если нужно

openFileDialog.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

openFileDialog.Title = "Открыть текстовый файл";

if (openFileDialog.ShowDialog() == DialogResult.OK)

{

try

{

string filePath = openFileDialog.FileName;

string fileContent = File.ReadAllText(filePath);

textInput.Text = fileContent; // Загружаем содержимое файла в textInput

}

catch (Exception ex)

{

MessageBox.Show($"Ошибка при открытии файла: {ex.Message}");

}

}

}

}

private void KMPSearch(string text, string pattern)

{

int m = pattern.Length;

int n = text.Length;

int[] lps = ComputeLPSArray(pattern);

int i = 0;

int j = 0;

while (i < n)

{

if (pattern[j] == text[i])

{

i++;

j++;

}

if (j == m)

{

MessageBox.Show($"Паттерн найден в индексе {i - j}n");

resultOutput.AppendText($"Паттерн найден в индексе {i - j}n");

j = lps[j - 1];

}

else if (i < n && pattern[j] != text[i])

{

if (j != 0)

j = lps[j - 1];

else

i++;

}

}

}

private int[] ComputeLPSArray(string pattern)

{

int length = 0;

int m = pattern.Length;

int[] lps = new int[m];

lps[0] = 0;

int i = 1;

while (i < m)

{

if (pattern[i] == pattern[length])

{

length++;

lps[i] = length;

i++;

}

else

{

if (length != 0)

length = lps[length - 1];

else

{

lps[i] = 0;

i++;

}

}

}

return lps;

}

private void BoyerMooreSearch(string text, string pattern)

{

int m = pattern.Length;

int n = text.Length;

// Создаем таблицу для плохих символов

int[] badChar = CreateBadCharacterTable(pattern);

int s = 0; // смещение паттерна

while (s <= n - m)

{

int j = m - 1;

// Сравниваем паттерн с текстом

while (j >= 0 && pattern[j] == text[s + j])

j--;

// Если паттерн найден

if (j < 0)

{

resultOutput.AppendText($"Паттерн найден в индексе {s}n");

s += (s + m < n) ? m - badChar[text[s + m]] : 1;

}

else

{

// Проверяем, чтобы не выйти за пределы массива badChar

int badCharIndex = (int)text[s + j];

if (badCharIndex >= 0 && badCharIndex < badChar.Length)

{

s += Math.Max(1, j - badChar[badCharIndex]);

}

else

{

s += 1; // Если символ не в диапазоне, просто сдвигаем на 1

}

}

}

}

private int[] CreateBadCharacterTable(string pattern)

{

int[] badChar = new int[256]; // Для всех возможных символов ASCII

for (int i = 0; i < badChar.Length; i++)

badChar[i] = -1; // Инициализируем таблицу значением -1

for (int i = 0; i < pattern.Length; i++)

badChar[(int)pattern[i]] = i; // Заполняем таблицу

return badChar;

}

private void button1\_Click(object sender, EventArgs e)

{

using (OpenFileDialog openFileDialog = new OpenFileDialog())

{

openFileDialog.InitialDirectory = @"C:\Downloads"; // Укажите начальную директорию, если нужно

openFileDialog.Filter = "Text files (\*.txt)|\*.txt|All files (\*.\*)|\*.\*";

openFileDialog.Title = "Открыть текстовый файл";

if (openFileDialog.ShowDialog() == DialogResult.OK)

{

try

{

string filePath = openFileDialog.FileName;

// Пробуем разные кодировки

string fileContent;

using (StreamReader reader = new StreamReader(filePath, Encoding.UTF8)) // Пробуйте разные кодировки

{

fileContent = reader.ReadToEnd();

}

textInput.Text = fileContent; // Загружаем содержимое файла в textInput

}

catch (Exception ex)

{

MessageBox.Show($"Ошибка при открытии файла: {ex.Message}");

}

}

}

}