Міністерство освіти і науки України

Івано-Франківський національний технічний університет нафти і газу

Кафедра ІПЗ

**Лабораторна робота № 10**

**«Методи пошуку рядків»**

Виконав

ст. гр. ІП-22-1

Микитій Ю.М.

Перевірив

доцент Процюк В.Р.

Івано-Франківськ

2023

**Мета:** Ознайомлення з основними методами пошуку, а також дослідження алгоритмів пошуку та вивчення способів їх реалізації за допомогою мови програмування С++.

**Завдання:**

1. Прямий метод пошуку

Нехай задамо масив S з "n" елементів та масив Р з "m" елементів(Студенти вибирають самостійно). Необхідно знайти перше входження масиву Р у масиві S.

2. Алгоритм Кнута, Моріса і Прата (КМП)

На вхід надходять два масиви символів:

S розміром n (текст) та інший масив Р розміром m (слово), які студент вибирає самостійно. Необхідно знайти перше входження слова у тексті.

Підготувати необхідний код мовою С++ для реалізації методів(відповідних)

**Розв’язок**

#include <locale>

#include <iostream>

#include <vector>

using namespace std;

// прямий метод пошуку

int search(string s, string p) {

int n = s.size();

int m = p.size();

for (int i = 0; i <= n - m; i++) {

int j;

for (j = 0; j < m; j++)

if (s[i + j] != p[j])

break;

if (j == m)

return i;

}

return -1;

}

// алгоритм кнута, моріса і прата (кмп)

vector<int> computeprefix(string p) {

int m = p.size();

vector<int> longestprefix(m);

for (int i = 1, k = 0; i < m; i++) {

while (k > 0 && p[k] != p[i])

k = longestprefix[k - 1];

if (p[k] == p[i])

k++;

longestprefix[i] = k;

}

return longestprefix;

}

int kmp(string s, string p) {

int n = s.size();

int m = p.size();

vector<int> longestprefix = computeprefix(p);

for (int i = 0, k = 0; i < n; i++) {

while (k > 0 && p[k] != s[i])

k = longestprefix[k - 1];

if (p[k] == s[i])

k++;

if (k == m)

return i - m + 1;

}

return -1;

}

int main() {

setlocale(LC\_CTYPE, "ukr");

// ваш код тут

string s = "hellhellow";

string p = "hellow";

int index = search(s, p);

if (index != -1) {

cout << "Прямий метод пошуку: слово знайдено в текстi на позицiї " << index << endl;

}

else {

cout << "Прямий метод пошуку: слово не знайдено в текстi" << endl;

}

index = kmp(s, p);

if (index != -1) {

cout << "Алгоритм кнута, морiса i прата (кмп): слово знайдено в текстi на позицiї " << index << endl;

}

else {

cout << "Алгоритм кнута, моріса i прата (кмп): слово не знайдено в текстi" << endl;

}

return 0;

}

**Результат**

