**Міністерство Освіти І НАУКИ України**

**Національний університет "Львівська політехніка"**

Інститут **КНІТ**

Кафедра **ПЗ**

### ЗВІТ

До лабораторної роботи № 10

**З дисципліни:** *“Алгоритми та структури даних”*

**На тему:** *“Простий пошук у стрічці”*

**Лектор:**

доц. каф. ПЗ

Коротєєва Т.О.

**Виконав:**

ст. гр. ПЗ-25

Морозов М.Ю.

**Прийняв:**

асист. каф. ПЗ

Цимбалюк Т.М.

« \_\_\_\_ » \_\_\_\_\_\_\_\_ 2018 р.

∑= \_\_\_\_\_ .

Львів – 2018

**Тема роботи:** Простий пошук у стрічці.

**Мета роботи:** Навчитися застосовувати алгоритм простого пошуку у стрічці при розв’язанні задач та перевірити його ефективність на різних наборах даних.

**TЕОРЕТИЧНІ ВІДОМОСТІ**

Алгори́тми по́шуку рядка́ (англ. string searching algorithms) — важливий клас рядкових алгоритмів, що намагаються знайти місце де один або декілька текстових рядків (зразків, англ. pattern) входять у довший рядок або текст.

Формальна постановка задачі пошуку рядка (англ. string-matching problem) така:

Нехай текст задано у вигляді масиву T[1..n] довжини n, а зразок — масиву P[1..m] довжини m≤n. Передбачається, що елементи масивів — символи із скінченного алфавіту Σ. Наприклад, алфавіт може мати вигляд Σ={0,1} чи Σ={a,b,…,z}.

Зразок P зустрічається у тексті T зі зсувом s (англ. occurs with shift s), якщо 0≤s≤n−m і T[s+1..s+m] = P[1..m] (іншими словами 1≤j≤m, T[s+j] = P[j]).

Якщо зразок P зустрічається у тексті T зі зсувом s, то величину s називають допустимим зсувом (англ. valid shift); інакше її називають недопустимим зсувом (англ. invalid shift)

Задача полягає в знаходженні всіх допустимих зсувів, з якими зразок P зустрічається у тексті T.

Алгоритм простого (прямого) пошуку у стрічці полягає в послідовній перевірці всіх елементів тексту T. При збігу T[i] з P[1] ми починаємо перевіряти інші символи P[1..m] з відповідними наступними символами T[i]. Якщо ми дойшли до P[m] і він збігся, то ми знайшли шуканий зразок у тексті. В противному випадку ідемо далі по тексту, виконуючи аналогічні дії. Цей алгоритм є найпростішим алгоритмом пошуку зразка в тексті. Його складність O(n \* m). Хоча він не є аніскілечки оптимізованим, я вважаю, що цей алгоритм чудово підходить як перший вивчений на предметі алгоритм пошуку зразка в стрічці, адже його не лише просто зрозуміти, а й запрограмувати.

**ЗАВДАННЯ**

Використовуючи алгоритм простого пошуку складіть програму, що шукає задане значення. Програма повинна забезпечити автоматичну генерацію масиву, виведення його на екран та покрокове виведення алгоритму пошуку.

*Варіант 9*

У заданому тексті знайти послідовність за шаблоном. У шаблоні число у фігурних дужках позначає кількість повторів символу чи групи (послідовності символів у круглих дужках) перед фігурною дужкою. Наприклад: шаблон bla{3}h позначає текст blaaah; шаблон (blah){3} позначає текст blahblahblah. Вивести шукану послідовність та всі її індекси в тексті.

**ХІД ВИКОНАННЯ**

Нехай задано масив S з n елементів та масив P з m елементів, m<=n. Необхідно знайти перше входження масиву P у масив S . Алгоритм зводиться до повтору порівнянь окремих елементів.

**Алгоритм R**

R1. Встановити і=1.. n-m, j=1..m.

R2. Якщо S[i] = P[j] , то зафіксувати перше співпадіння k=i, та перевірити співпадіння всього масиву P у масиві S. При першому неспівпадінні відмінити значення k та продовжити пошук.

R3. Кінець.

Функція, яка шукає зразок в тексті методом простого пошуку у стрічці:

public static void SimpleSearch (string inputArray, string key)

{

for(int j = 0; j < inp.Length; ++j)

{

if (inp[j] == key[0]) //found first symbol of key

{

steps.Text+="Found first symbol "+inp[j]+" at index: "+j.ToString() + "\n";

++j;

for (int k = 1; k < key.Length; ++k, ++j) {

if (inp[j] != key[k]) {

steps.Text += "Failed match of key at index: " + (j + k).ToString() + "\n";

break; }

if (k == key.Length - 1) {

steps.Text += "Found key at index: " + j.ToString() + "\n";

res += "Found key at index: " + j.ToString() + "\n";

++cnt; }}}}}

**РЕЗУЛЬТАТИ**

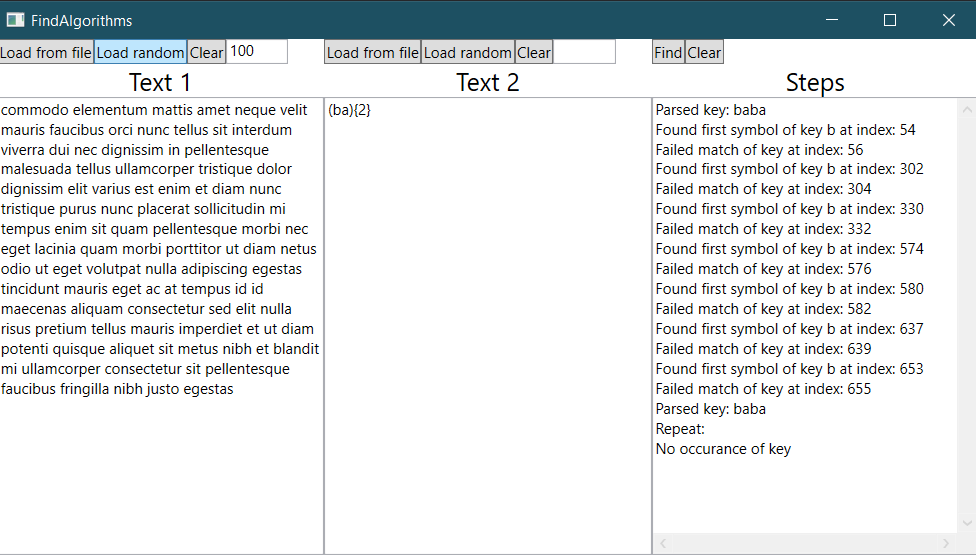
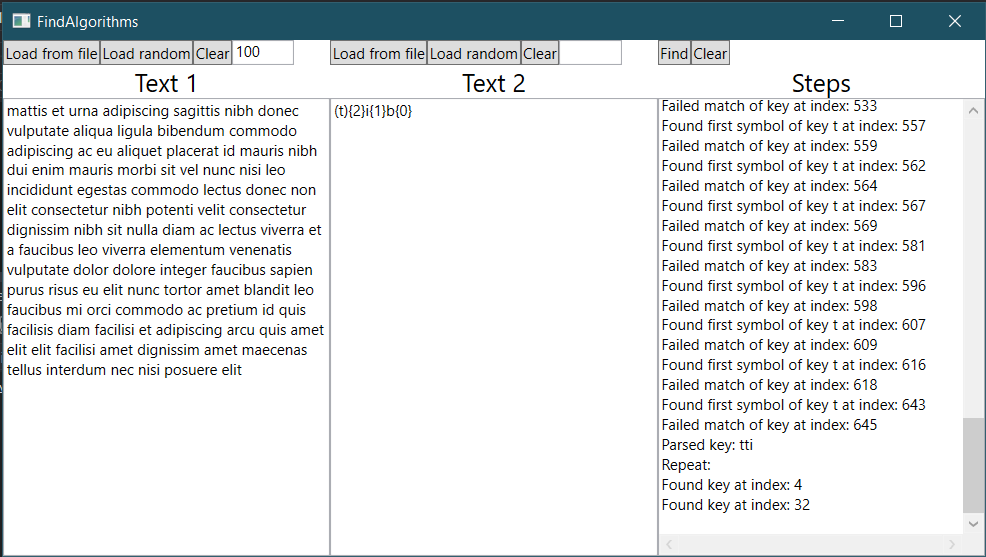


Рис.1. Провальний пошук baba

Рис.2. Успішний пошук tti

**ВИСНОВКИ**

На даній лабораторній роботі я розглядав реалізацію алгоритму простого пошуку стрічки. В результаті виконання лабораторної роботи я навчився використовувати даний алгоритм для пошуку елементів з вказаними властивостями, зокрема навчився перетворювати закодований ключ в звичайний згідно свого індивідуального завдання. Варто зазначити, що даний алгоритми не є ефективним, тому не застосовується на практиці. Обчислювальна складність - **.**