Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина»

Кафедра КТ

Отчет о лабораторной работе № 3

Реализация алгоритма

“Алгоритмы и структуры данных”

Выполнил:

Студент группы 948

Голдина Н.А.

Проверил:

доцент каф. КТ

к.т.н. Наумов Д.А.

Рязань 2021

**Задание (вариант 2)**

|  |
| --- |
| Подстроки |
|  | Входной файл состоит из одной строки I, содержащей малые буквы английского алфавита. |
|  | Назовём подстрокой длиной L с началом S множество непрерывно следующих символов строки. | |
|  | Например, строка | |
|  |  | |
|  | abcab | |
|  | содержит подстроки | |
|  |  | |
|  | длины 1: a, b, c, a, b | |
|  | длины 2: ab, bc, ca, ab | |
|  | длины 3: abc, bca, cab | |
|  | длины 4: abca, bcab | |
|  | длины 5: abcab | |
|  |  | |
|  | В строках длины 1 есть два повторяющихся элемента - a и b. Назовём весом подстрок длины L произведение максимального количества повторяющихся подстрок этой длины на длину L. | |
|  | В нашем случае вес длины 1 есть 2 (2·1), длины 2 есть 4 (2·2), длины 3 - 3 (1·3), длины 4 - 4 и длины 5 - 5. | |
|  | Требуется найти наибольший из всех весов различных длин. | |
|  | Input format | |
|  | Output format | |
|  | Examples | |
|  | Input Output | |
|  | aabaabaabaabaa | |
|  | 24 | |
|  | abcab | |
|  | 5 | |
|  | \*/ | |
|  |  | |
|  | #include <iostream> | |
|  | #include <string> | |
|  | #include <map> | |
|  |  | |
|  | using namespace std; | |
|  |  | |
|  | int main () { | |
|  | string s; | |
|  | cin >> s; | |
|  |  | |
|  | int max = 0; | |
|  |  | |
|  | map <string, int> m; | |
|  | for (int len = 1; len<=s.length(); len++) { | |
|  | for (int i=0; i<=s.length()-len; i++) { | |
|  | if (m[s.substr(i, len)]) {m[s.substr(i, len)]++;} | |
|  | else {m[s.substr(i, len)] = 1;} | |
|  | if (m[s.substr(i, len)]\*len > max) {max = m[s.substr(i, len)]\*len;} | |
|  | } | |
|  | } | |
|  |  | |
|  | cout << max; | |
|  |  | |
|  | return 0; | |
|  | } | |