**Министерство образования и науки Российской Федерации**

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,   
МЕХАНИКИ И ОПТИКИ**

Факультет программной инженерии и компьютерной техники  
Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Aлгоритмы и структуры данных»

**ОТЧЁТ**

по лабораторной работе №10 (Week 10 Openedu)

Студент Дунаев Алексей Игоревич

Группа P3217

Преподаватель Муромцев Дмитрий Ильич

Санкт-Петербург

2019 г.

# Задача 1 Префикс-функция

|  |  |
| --- | --- |
| Имя входного файла: | input.txt |
| Имя выходного файла: | output.txt |
| Ограничение по времени: | 2 секунды |
| Ограничение по памяти: | 256 мегабайт |

Постройте префикс-функцию для всех непустых префиксов заданной строки .

#### Формат входного файла

Первая строка входного файла содержит  . Строка состоит из букв латинского алфавита.

#### Формат выходного файла

Выведите значения префикс-функции для всех префиксов строки  длиной , в указанном порядке.

#### Примеры

|  |  |
| --- | --- |
| input.txt | output.txt |
| aaaAAA | 0 1 2 0 0 0 |
| abacaba | 0 0 1 0 1 2 3 |

## Исходный код к задаче 1

#include **<vector>**

#include **<iostream>**

#include **<algorithm>**

#include **<string>**

#include **<unordered\_map>**

#include **<list>**

**using namespace** std;

*/\*\*/*

#include **"edx-io.hpp"**

#define **cout** io

#define **cin** io

*/\*\*/*

vector<**int**> prefix\_function (string s) {

**int** n = (**int**) s.length();

vector<**int**> pi (n);

**for** (**int** i=1; i<n; ++i) {

**int** j = pi[i-1];

**while** (j > 0 && s[i] != s[j])

j = pi[j-1];

**if** (s[i] == s[j]) ++j;

pi[i] = j;

}

**return** pi;

}

**int** main() {

string s;

**cin** >> s;

**auto** v = prefix\_function(s);

**for** (vector<**int**>::iterator i = v.begin(); i != v.end(); ++i)

{

**cout** << \*i << **" "**;

}

**cout** << **"\n"**;

**return** 0;

}

## Бенчмарк к задаче 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № **теста** | **Результат** | **Время, с** | **Память** | **Размер входного файла** | **Размер выходного файла** |
| Max |  | 0.281 | 10416128 | 1000002 | 6888892 |
| 1 | OK | 0.000 | 3448832 | 8 | 14 |
| 2 | OK | 0.000 | 3452928 | 9 | 16 |
| 3 | OK | 0.000 | 3440640 | 3 | 4 |
| 4 | OK | 0.015 | 3436544 | 4 | 6 |
| 5 | OK | 0.000 | 3436544 | 4 | 6 |
| 6 | OK | 0.015 | 3473408 | 12 | 22 |
| 7 | OK | 0.000 | 3461120 | 12 | 22 |
| 8 | OK | 0.015 | 4030464 | 92672 | 185342 |
| 9 | OK | 0.046 | 4116480 | 99998 | 588848 |
| 10 | OK | 0.015 | 4091904 | 100002 | 561031 |
| 11 | OK | 0.046 | 4636672 | 176391 | 352780 |
| 12 | OK | 0.062 | 4796416 | 199994 | 1288781 |
| 13 | OK | 0.046 | 4792320 | 199992 | 1190919 |
| 14 | OK | 0.031 | 4603904 | 172864 | 345726 |
| 15 | OK | 0.093 | 5464064 | 300002 | 1988892 |
| 16 | OK | 0.062 | 5472256 | 300002 | 1716601 |
| 17 | OK | 0.062 | 5169152 | 249367 | 498732 |
| 18 | OK | 0.093 | 6201344 | 400002 | 2688857 |
| 19 | OK | 0.093 | 6197248 | 399998 | 2333026 |
| 20 | OK | 0.093 | 6561792 | 455342 | 910682 |
| 21 | OK | 0.156 | 6901760 | 499996 | 3388800 |
| 22 | OK | 0.109 | 6897664 | 499998 | 2875818 |
| 23 | OK | 0.109 | 6942720 | 505139 | 1010276 |
| 24 | OK | 0.187 | 7593984 | 600000 | 4088813 |
| 25 | OK | 0.156 | 7593984 | 600002 | 3977737 |
| 26 | OK | 0.156 | 7155712 | 539096 | 1078190 |
| 27 | OK | 0.171 | 8298496 | 699998 | 4788809 |
| 28 | OK | 0.171 | 8310784 | 700002 | 4566502 |
| 29 | OK | 0.093 | 6828032 | 492073 | 984144 |
| 30 | OK | 0.203 | 8990720 | 799997 | 5488767 |
| 31 | OK | 0.187 | 9015296 | 800002 | 3984420 |
| 32 | OK | 0.156 | 8744960 | 763540 | 1527078 |
| 33 | OK | 0.203 | 9703424 | 899994 | 6188746 |
| 34 | OK | 0.218 | 9670656 | 900002 | 5662195 |
| 35 | OK | 0.171 | 9256960 | 836144 | 1672286 |
| 36 | OK | 0.281 | 10375168 | 1000002 | 6888857 |
| 37 | OK | 0.234 | 10391552 | 1000002 | 6555317 |
| 38 | OK | 0.218 | 10412032 | 1000002 | 2000002 |
| 39 | OK | 0.234 | 10391552 | 1000002 | 6888892 |
| 40 | OK | 0.234 | 10387456 | 1000002 | 6888892 |
| 41 | OK | 0.250 | 10391552 | 1000002 | 6209775 |
| 42 | OK | 0.234 | 10379264 | 1000002 | 6888887 |
| 43 | OK | 0.265 | 10391552 | 1000002 | 6888872 |
| 44 | OK | 0.234 | 10399744 | 1000002 | 6388897 |
| 45 | OK | 0.234 | 10416128 | 1000002 | 6883897 |

# Задача 2. Z-функция

|  |  |
| --- | --- |
| Имя входного файла: | input.txt |
| Имя выходного файла: | output.txt |
| Ограничение по времени: | 2 секунды |
| Ограничение по памяти: | 256 мегабайт |

Постройте Z-функцию для заданной строки .

#### Формат входного файла

Первая строка входного файла содержит  . Строка состоит из букв латинского алфавита.

#### Формат выходного файла

Выведите значения Z-функции для всех индексов  строки , в указанном порядке.

#### Примеры

|  |  |
| --- | --- |
| input.txt | output.txt |
| aaaAAA | 2 1 0 0 0 |
| abacaba | 0 1 0 3 0 1 |

## Исходный код к задаче 2

**#include <vector>**

#include **<iostream>**

#include **<algorithm>**

#include **<string>**

#include **<unordered\_map>**

#include **<list>**

**using namespace** std;

*/\*\*/*

#include **"edx-io.hpp"**

#define **cout** io

#define **cin** io

*/\*\*/*

vector<**int**> calc\_z(string s) {

vector<**int**> z(s.length(), 0);

**for** (**int** i = 1, l = 0, r = 0; i < s.length(); i++) {

**if** (i <= r) {

z[i] = min(z[i - l], r - i + 1);

}

**while** (i + z[i] < s.length()) {

**if** (s[i + z[i]] == s[z[i]]) {

z[i]++;

} **else** {

**break**;

}

}

**if** (z[i] > 0 && i + z[i] - 1 > r) {

l = i;

r = i + z[i] - 1;

}

}

**return** z;

}

**int** main() {

string s;

cin >> s;

**auto** v = calc\_z(s);

**for** (vector<**int**>::iterator i = v.begin()+1; i != v.end(); ++i)

{

cout << \*i << **" "**;

}

cout << **"\n"**;

**return** 0;

}

## Бенчмарк к задаче 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № **теста** | **Результат** | **Время, с** | **Память** | **Размер входного файла** | **Размер выходного файла** |
| Max |  | 0.281 | 10412032 | 1000002 | 6888890 |
| 1 | OK | 0.000 | 3444736 | 8 | 12 |
| 2 | OK | 0.046 | 3457024 | 9 | 14 |
| 3 | OK | 0.000 | 3444736 | 4 | 4 |
| 4 | OK | 0.000 | 3448832 | 4 | 4 |
| 5 | OK | 0.015 | 3436544 | 5 | 6 |
| 6 | OK | 0.000 | 3477504 | 12 | 20 |
| 7 | OK | 0.000 | 3452928 | 12 | 20 |
| 8 | OK | 0.031 | 4030464 | 92672 | 185340 |
| 9 | OK | 0.031 | 4083712 | 99998 | 264803 |
| 10 | OK | 0.015 | 4083712 | 100002 | 272213 |
| 11 | OK | 0.031 | 4640768 | 176391 | 352778 |
| 12 | OK | 0.046 | 4784128 | 199994 | 474052 |
| 13 | OK | 0.046 | 4800512 | 199992 | 456481 |
| 14 | OK | 0.031 | 4628480 | 172864 | 345724 |
| 15 | OK | 0.078 | 5472256 | 300002 | 1988890 |
| 16 | OK | 0.078 | 5480448 | 300002 | 786115 |
| 17 | OK | 0.062 | 5144576 | 249367 | 498730 |
| 18 | OK | 0.109 | 6176768 | 400002 | 1036110 |
| 19 | OK | 0.093 | 6180864 | 399998 | 885170 |
| 20 | OK | 0.093 | 6553600 | 455342 | 910680 |
| 21 | OK | 0.109 | 6889472 | 499996 | 1217155 |
| 22 | OK | 0.109 | 6881280 | 499998 | 1267988 |
| 23 | OK | 0.109 | 6922240 | 505139 | 1010274 |
| 24 | OK | 0.125 | 7589888 | 600000 | 1406342 |
| 25 | OK | 0.125 | 7577600 | 600002 | 1477778 |
| 26 | OK | 0.156 | 7180288 | 539096 | 1078188 |
| 27 | OK | 0.140 | 8269824 | 699998 | 1682395 |
| 28 | OK | 0.187 | 8290304 | 700002 | 1558333 |
| 29 | OK | 0.093 | 6840320 | 492073 | 984142 |
| 30 | OK | 0.203 | 9007104 | 799997 | 1804664 |
| 31 | OK | 0.171 | 8974336 | 800002 | 2196118 |
| 32 | OK | 0.156 | 8736768 | 763540 | 1527076 |
| 33 | OK | 0.218 | 9687040 | 899994 | 2030973 |
| 34 | OK | 0.187 | 9674752 | 900002 | 2765566 |
| 35 | OK | 0.171 | 9224192 | 836144 | 1672284 |
| 36 | OK | 0.218 | 10379264 | 1000002 | 2611110 |
| 37 | OK | 0.218 | 10403840 | 1000002 | 2227777 |
| 38 | OK | 0.203 | 10391552 | 1000002 | 2000000 |
| 39 | OK | 0.281 | 10412032 | 1000002 | 6888890 |
| 40 | OK | 0.281 | 10395648 | 1000002 | 6888890 |
| 41 | OK | 0.218 | 10379264 | 1000002 | 2841973 |
| 42 | OK | 0.218 | 10371072 | 1000002 | 4444445 |
| 43 | OK | 0.250 | 10399744 | 1000002 | 2977778 |
| 44 | OK | 0.203 | 10403840 | 1000002 | 2000045 |
| 45 | OK | 0.203 | 10379264 | 1000002 | 2004887 |

# 