



**Факультет программной инженерии и
компьютерной техники**

Алгоритмы и структуры данных

Лабораторная работа №3

Сортировка

Преподаватель: Косяков Михаил Сергеевич

Выполнил: Ле Чонг Дат

Группа: P3231

2021 г.

1 Задание

Вариант: 1322. Шпион

На вход устройства подается строка текста $S_1 = s_1s_2s_3...s_n$. Получив ее, устройство строит все циклические перестановки этой строки, то есть $S_2 = s_2s_3...s_ns_1, ..., S_N = s_ns_1...s_{n-1}$. Затем множество строк $S_1, S_2, ..., S_N$ сортируется лексикографически по возрастанию. И в этом порядке строчки выписываются в столбец, одна под другой. Получается таблица размером $N * N$. В какой-то строке K этой таблицы находится исходное слово. Номер этой строки вместе с последним столбцом устройство и выдает на выход.

Например, если исходное слово $S_1 = \text{abracadabra}$, то таблица имеет такой вид:

1. aabracadabr = S_{11}
2. abraabracad = S_8
3. abracadabra = S_1
4. acadabraabr = S_4
5. adabraabrac = S_6
6. braabracada = S_9
7. bracadabraa = S_2
8. cadabraabra = S_5
9. dabraabraca = S_7
10. raabracadab = S_{10}
11. racadabraab = S_3

И результатом работы устройства является число 3 и строка rdarcaaabb.

2 Решение

2.1 Алгоритм

Легко видеть, что символы в последнем столбце также являются символами исходной строки. Поскольку строки отсортированы лексикографически, когда мы располагаем символы в последнем столбце, мы получаем первый столбец в таблице.

Посмотрим, что у нас есть сейчас. Для примера в задаче:

1. a????????r
2. a????????d
3. a????????a
4. a????????r

5. a????????c
6. b????????a
7. b????????a
8. c????????a
9. d????????a
10. r????????b
11. r????????b

l_i : позиция строки в таблице, полученная после поворота i -й строки в таблице влево. Если $i < j$, то $l_i < l_j$.

→ Для строки i , l_i будет первой позицией в таблице, которая не соответствует ни одной предыдущей строке и, возможно, строке i после поворота влево.

Для примера в задаче:

1. a????????r → $l_i = 3$
2. a????????d → $l_i = 6$
3. a????????a → $l_i = 7$
4. a????????r → $l_i = 8$
5. a????????c → $l_i = 9$
6. b????????a → $l_i = 10$
7. b????????a → $l_i = 11$
8. c????????a → $l_i = 5$
9. d????????a → $l_i = 2$
10. r????????b → $l_i = 1$
11. r????????b → $l_i = 4$

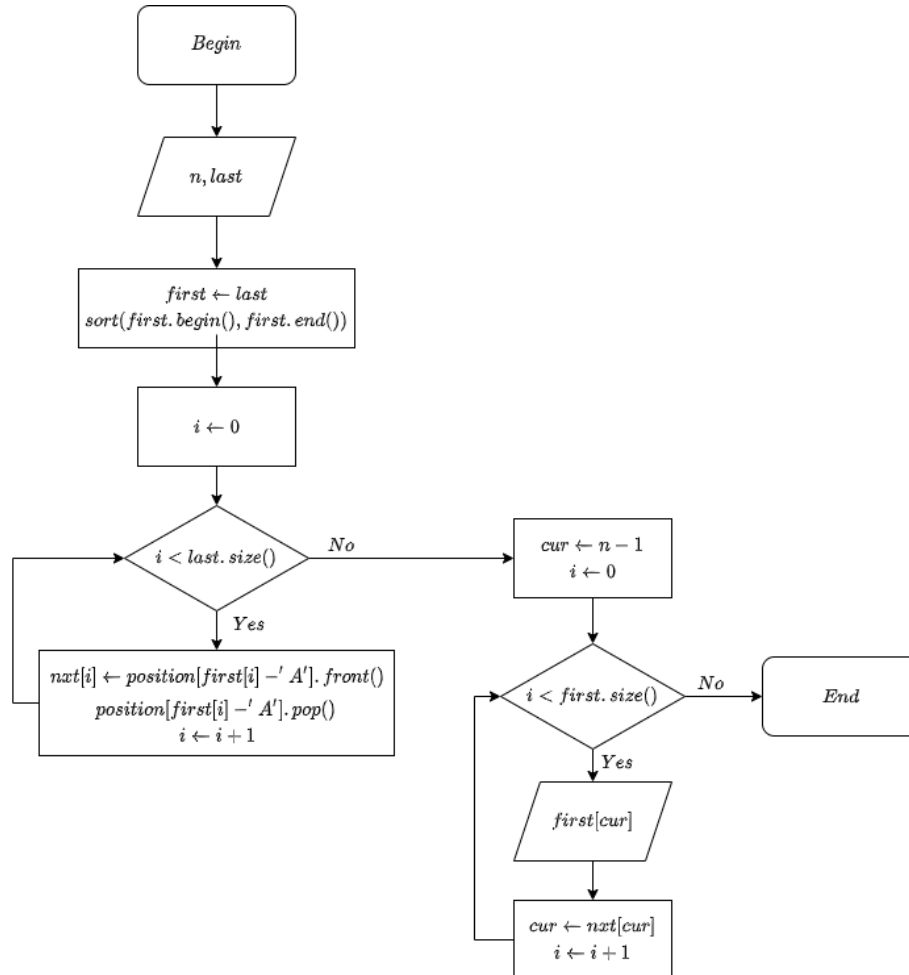
Чтобы найти массив l_i , мы используем очередь $position_i$, где $position_i$ - позиции строк, заканчивающихся с i .

3 Схема программы

Глобальные переменные:

- n - Позиция исходного строки в таблице
- $first$ - Первый столбец
- $last$ - Последний столбец
- $position_i$ - Позиции строк, заканчивающихся с i .

- nxt_i - Позиция строки в таблице, полученная после поворота i -й строки в таблице влево



4 Реализация

```

#include <bits/stdc++.h>
using namespace std;
const int N = 100010;
const int M = 'z' - 'A' + 1;
int n;
string first, last;
queue<int> position[M];
int nxt[N];
int main() {
    ios::sync_with_stdio(0);
    cin.tie(0); cout.tie(0);

    cin >> n;

```

```

cin >> last;
first = last;
sort(first.begin(), first.end());
memset(nxt, -1, sizeof nxt);
for(int i = 0; i < last.size(); ++i) {
    position[last[i] - 'A'].push(i);
}
for(int i = 0; i < first.size(); ++i) {
    nxt[i] = position[first[i] - 'A'].front();
    position[first[i] - 'A'].pop();
}
for(int cur = n - 1, i = 0; i < first.size(); ++i) {
    cout << first[cur];
    cur = nxt[cur];
}
}

```

5 Вывод

Сортировка - чрезвычайно распространенный алгоритм в нашей жизни. Существует множество алгоритмов сортировки, таких как пузырьковая сортировка, сортировка вставкой, сортировка по куче и т.д . Я научился использовать сортировку для эффективного инвертирования преобразования Барроуза-Уиллера.