

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МО ЭВМ**

**ОТЧЕТ**  
**по лабораторной работе №2**  
**по дисциплине «Алгоритмы и структуры данных»**  
**Тема: "Иерархические списки"**

Студентка гр. 9383

Ноздрин В.Я.

Преподаватель

Попова Е.В.

Санкт-Петербург

2020

### **Цель работы.**

Ознакомиться с основными понятием иерархического списка, решить задачу связанную с иерархическими списками.

### **Основные теоретические положения.**

Рекурсия – способ организации вычислительного процесса, при котором процедура или функция в ходе выполнения составляющих ее операторов обращается сама к себе.

Иерархический список представляет собой или элемент базового типа E1, называемый в этом случае атомом (атомарным S-выражением), или линейный список из S-выражений.

Традиционно иерархические списки представляют или графически, используя для изображения структуры списка двухмерный рисунок, или в виде одномерной скобочной записи.

### **Задание.**

Вариант №13

Вычислить глубину (число уровней вложения) иерархического списка как максимальное число одновременно открытых левых скобок в сокращенной скобочной записи списка; принять, что глубина пустого списка и глубина атомарного S-выражения равны нулю; например, глубина списка (a (b ( ) c) d) равна двум.

### **Ход работы:**

1. Провести анализ задачи и выбрать подход к решению.
2. Разработать программу.
3. Протестировать программу.

### **Выполнение работы:**

Так как в данной задаче нам дается правильная скобочная запись иерархического списка, было принято решение решать ее динамически.

В функции **main** файл **input.txt** считывается построчно и для каждой строки вызывается функция **depth**, которая возвращает целочисленное значение, равное глубине списка.

Функция **depth** принимает на вход строку и возвращает глубину списка. Она выполняет проход по строке и считает открытые и закрытые скобки, игнорируя “пустые” скобки и атомарные литералы.

### **Пример работы программы:**

#### **Входные данные (в файле input.txt):**

(a(b()c)d)

(a(b()(s(nine)))(ds)c)d)

(a(b()c)d)

#### **Выходные данные:**

depth of "(a(b()c)d)" is 2

depth of "(a(b()(s(nine)))(ds)c)d)" is 4

depth of "(a(b()c)d)" is 2

Разработанный программный код см. в приложении А.

### **Вывод.**

Было произведено знакомство с устройством иерархических списков, а так разработана программа для подсчета глубины иерархических списков в линейной скобочной записи. Код написан на языке программирования C++.

## Приложение А

### Исходный код программы

Название файла: main.c

```
#include <iostream>
#include <fstream>
#include <string>

int depth(std::string str) {
    int result = 0;
    int opened = 0;

    for(std::string::iterator it=str.begin(); it!=str.end();it++)
    {
        switch (*it) {
            case '(': {
                if (*(it+1) != ')') {
                    opened += 1;
                } else {
                    it++;
                }
                break;
            }
            case ')': {
                if (opened > result)
                    result = opened;
                opened--;
            }
            default: {
                break;
            }
        }
    }

    return result;
}

int main() {
    std::ifstream input("input.txt");
    for (std::string line; getline(input, line); ) {
        std::cout << "depth of \"" << line << "\" is " <<
depth(line) << "\n";
    }

    return 0;
}
```