

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

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Алгоритмы и структуры данных»
Тема: Рекурсия

Студент гр. 9382

Докукин В.М.

Преподаватель

Фирсов М.А.

Санкт-Петербург

2020

Цель работы.

Изучить понятие рекурсии, способы применения рекурсии для решения задач и её реализации на языке программирования C++.

Основные теоретические положения.

Рекурсией в общем случае называется определение, описание, изображение какого-либо объекта или процесса внутри самого этого объекта или процесса, то есть ситуация, когда объект является частью самого себя. В более узком смысле рекурсия — это выражение функции через саму себя, т. е. вызов функцией самой себя внутри своего тела.

Задание.

7. Построить синтаксический анализатор для понятия *вещественное число*.

*вещественное_число ::= целое_число . целое_без_знака /
целое_число . целое_без_знака* *Е* *целое_число /
целое_число*
целое_без_знака ::= цифра | цифра целое_без_знака
целое_число ::= целое_без_знака / + целое_без_знака / - целое_без_знака

Ход работы.

1. Определена рекурсивная часть задачи: число рекурсивно определяется как целое без знака, если оно состоит из цифры или цифры и целого без знака.

2. Написаны 3 функции:

1) `bool isReal(std::string str)` - функция, которая определяет, является ли число вещественным. Для этого входная строка разбивается на 1-3 подстроки (в зависимости от того, имеются ли в ней символы «.» и «Е»), которые направляются в функции `isSignedInt()` и `isUnsignedInt()`.

2) `bool isSignedInt(std::string str)` — функция, определяющая, является ли число знаковым целым. Основное предназначение — проверка знака числа. Функция вернёт `false`, если обнаружит некорректность записи числа, или значение `isUnsignedInt()`, если ошибок не обнаружится.

3) `bool isUnsignedInt(std::string str)` – рекурсивная функция, которая проверяет, является ли входная строка числом. Рекурсия заканчивается, когда находится ошибка формата или в строке остаётся один символ-цифра.

3. В функции `main()` осуществлён ввод данных в программу из файла `tests.txt` и вывод результата обработки входных строк в консоль, а также имеется возможность ввести свои данные через консоль.

Пример работы программы.

Входные данные	Выходные данные
1234.567E89	1234.567E89 dotpos = 4 epos = 8 str1 = 1234, str2 = 567, str3 = 89 Recursion depth: 1, symbol to check: 2 Recursion depth: 2, symbol to check: 3 Recursion depth: 3, symbol to check: 4 Recursion depth: 1, symbol to check: 5 Recursion depth: 2, symbol to check: 6 Recursion depth: 3, symbol to check: 7 Recursion depth: 1, symbol to check: 9 1234.567E89 is a real number.

Тестирование.

Результаты тестирования представлены в таблице ниже.

№ теста	Входные данные	Выходные данные	Комментарий
1	98E765.4321	98E765.4321 dotpos = 6 epos = 2 98E765.4321 is not a real number.	Так как символ «.» расположен дальше «E», программа досрочно завершает обработку и выдаёт сообщение о том, что строка числом не является.
2	2222.22.222	2222.22.222 dotpos = 4 epos = -1 str1 = 2222, str2 = 22.222, str3 = 0 Recursion depth: 1, symbol to check: 2 Recursion depth: 2, symbol to check: 2 Recursion depth: 3, symbol to check: 2 Recursion depth: 1, symbol to check: 2	Проверка взаимодействия программы со строкой, содержащей множество вхождений

		Recursion depth: 2, symbol to check: 2 Recursion depth: 3, symbol to check: . 2222.22.222 is not a real number.	символа «.».
3	c++	c++ dotpos = -1 epos = -1 str1 = c++, str2 = 0, str3 = 0 Recursion depth: 1, symbol to check: 0 c++ is not a real number.	Проверка взаимодействия программы со строкой, не являющейся числом.
4	[пустая строка]	dotpos = -1 epos = -1 is not a real number.	Проверка взаимодействия программы с пустой строкой.

Выводы.

В результате выполнения лабораторной работы были:

1. Изучены понятие рекурсии, способы её применения и реализации на языке программирования C++.
2. Написана программа, решающая поставленную задачу.
3. Написана серия тестов, позволяющих качественно оценить работу программы (тесты находятся в файле tests.txt).

Код программы размещён в Приложении 1.

ПРИЛОЖЕНИЕ 1

ИСХОДНЫЙ КОД ПРОГРАММЫ

Имя файла: main.cpp

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

bool isUnsignedInt(std::string str, int depth = 0){ //Проверяем,
является ли целым беззнаковым ( РЕКУРСИВНАЯ: целое_беззнаковое :=
цифра || цифра целое_беззнаковое )
    for (int i = 0; i < depth; i++){
        std::cout<<" ";
    }
    std::cout<<"Recursion depth: "<<depth+1<<" , symbol to check:
"<<str[0]<<'\n';
    if ( (int)str[0] > 47 && (int)str[0] < 58 ){
        if ( str.length() == 1 ) return true;
        else return isUnsignedInt(str.substr(1, str.length() - 1),
++depth);
    }
    else return false;
}

bool isSignedInt(std::string str){ //Проверяем, является ли целым
знаковым ( целое := целое_беззнаковое || +целое_беззнаковое || -
целое_беззнаковое )
    if ( (str[0] == '+' || str[1] == '-') && str.length() == 1 )
return false; // Если
в строке только "+" или "-" - неверный формат.
    if ( (int)str[0] > 47 && (int)str[0] < 58 && str.length() ==
1 ) return true;

    if ( (str[0] == '+' || str[1] == '-') && str.length() > 1 )
return isUnsignedInt(str.substr(1, str.length() - 1)); // Если в
строке "+" или "-"
    if ( (int)str[0] > 47 && (int)str[0] < 58 ) return
isUnsignedInt(str.substr(1, str.length() - 1));
// или цифра на 1 позиции, передаём строку без первого символа в
isUnsignedInt
}

bool isReal(std::string str){ // Проверяем, является ли
вещественным ( Вещественное := целое.целое_беззнаковое ||
целое.целое_беззнаковоеЕцелое || целоеЕцелое )
    int dotpos = str.find("."), epos = str.find("E");
    std::string str1, str2, str3;
    int npos = std::string::npos;
    std::cout<<"dotpos = "<<dotpos<<' ' <<"epos = "<<epos<<'\n';
```

```

        if ( dotpos > epos && epos != npos ) return false;
                                // Если есть и точка, и Е, и точка идёт
после Е - неверный формат.
        if ( dotpos == 0 && epos == 0)                return false;
                                // Если точка или Е в самом начале -
неверный формат.
        if ( dotpos + 1 == epos )                return false;
                                // Если между точкой и Е ничего нет -
неверный формат.
        if ( dotpos == str.length() - 1 || epos == str.length() - 1 )
return false; // Если после точки или Е идёт конец строки -
неверный формат.
        //Разбиваем на подстроки
        if ( dotpos == npos && epos == npos ) str1 = str;
        if ( dotpos == npos && epos != npos ) str1 = str.substr(0,
epos);
        if ( dotpos != npos ) str1 = str.substr(0, dotpos);
        if ( dotpos == npos ) str2 = "0";
        if ( dotpos != npos && epos == npos ) str2 =
str.substr(dotpos + 1, str.length() - dotpos - 1);
        if ( dotpos != npos && epos != npos ) str2 =
str.substr(dotpos + 1, epos - dotpos - 1);
        if ( epos == npos ) str3 = "0";
        if ( epos != npos ) str3 = str.substr(epos + 1, str.length()
- epos - 1);
        std::cout<<"str1 = "<<str1<<"", str2 = "<<str2<<"", str3 =
"<<str3<<'\n'";
        return isSignedInt(str1) & isUnsignedInt(str2) &
isSignedInt(str3);
    }

int main(){
    std::string str;
    int a = 0; // Выбор опции ввода

    std::ifstream f("tests.txt");

    std::cout<<"Choose input option(0 - file input; 1 - console
input)\n";
    std::cin>>a;
    if (a){
        std::cin>>str;
        if (isReal(str)){
            std::cout<<str<<" is a real number.\n";
        }
        else{
            std::cout<<str<<" is not a real number.\n";
        }
        return 0;
    }
    if (!f){

```

```

        std::cout<<"Couldn't open file!\n";
        return 0;
    }
    while(!f.eof()){
        getline(f, str);
        std::cout<<str<<'\\n';
        if (isReal(str)){
            std::cout<<str<<" is a real number.\n";
        }
        else{
            std::cout<<str<<" is not a real number.\n";
        }
        std::cout<<"-----\n";
    }
    return 0;
}

```