Министерство образования и науки Российской Федерации

*Санкт-Петербургский государственный электротехнический университет*

**Центр Новых Информационных Технологий**

**Кафедра МОЭВМ.**

**Отчёт по лабораторной работе:**

## **«Индуктивные функции»**

Преподаватель: Самойленко В.П.

Выполнили: студенты гр.5383

Десятов А.С.

Допира В. Е.

Санкт-Петербург

2015

**Лабораторная работа 5**

**Индуктивные функции**

**1.Задача**

Заданы алфавит X и функция f: Ω(X) →Y.

1. Определить, является ли функция f(ω) индуктивной. Если “да” – выписать функцию G(\*,\*), если “нет” – подтвердить это, применяя критерий индуктивности, а затем придумать индуктивное расширение и выписать G для него.

2. Написать программу вычисления указанной функции f(ω), считая, что ω – это состояние последовательности элементов типа X, находящейся во входном файле.

**4.** Число локальных максимумов, f: Ω(R) →N0. (Элемент последовательности называется локальным максимумом, если у него нет соседа, большего, чем сам элемент. Например, в любой одноэлементной последовательности ровно 1 локальный максимум.)

**2.Содержательная постановка задачи**

Проверить функцию на индуктивность.

f(x0,x1…xn)

max(x0,x1,…,xn)=xj, где ∀i∈{0,1,…,n}⇒xj≥xi.

Эта функция индуктивна, поскольку max(ω∟x)={если x>max(ω), то x, иначе max(ω).

Таким образом, F(f,x)={если x>f, то x, иначе f}=max(f,x).

Максимумум должен быть не больше любого из х.

max(ω∟x)={еслиx>max(ω), то x, иначе max(ω).

Программа использует данные из файла. Она ищет локальные максимумы среди последовательных трех элементов и выводит в другой файл их количество. Также должны выводить поочередно обрабатываемые элементы и промежуточные значения.

Пример решения: x0=1.2 х1=0.5

1.2 > 0.5, значит 1.2=max. Других переменных нет, значит локальный максимум один.

**3.Алгоритм**

**Входные данные – последовательность х, заданная в файле.**

**Выходные данные – количество локальных максимумов i.**

**Переменные: Вещественные x0 , x1,x2; Целые: i.**

**Шаг 1:** Программа берет данных из файла "in\_data.txt".

**Шаг 2:** Если в папке нет файла с таким именем, программа выводит сообщение "Входной файл не открыт!".

**Шаг 3:** Проверка первого элемента на локальный максимум. Если в файле нет переменных, водится сообщение "Файл пуст!".

**Шаг 4:** Если в файле есть только одно значение х0, то оно и является локальным максимумом. Если два- то программа ищет максимум среди этих значений.

**Шаг 5:** Далее идет проверка на наличие третьего значения и поиск локальных максимумов среди всех значений.

**Шаг 6:** Программа работает пока в файле есть значения х.

**Шаг 7:** Последний элемент сравнивается с предыдущим. Если крайний оказывается больше, то он тоже является локальным максимумом.

**Шаг 8:** После завершения поиска максимумов конечное значение записывается в файл "out\_data.txt".

**4. Набор тестов**

|  |  |  |
| --- | --- | --- |
| **Входные данные** | **Ожидаемый результат** | **Полученный результат** |
| **Локальный максимум** | **Локальный максимум** |
| В папке нет файла | Входной файл не открыт! | Входной файл не открыт! |
| В файле нет данных | Файл пуст! | Файл пуст! |
| 1.2 | 1 | 1 |
| 1.2 0.5 | 1 | 1 |
| 0.5 1.2 1.7 | 1 | 1 |
| 0.5 1.2 1 | 1 | 1 |
| 3.1 -5. 11.2 -13.3 7 5.9 1.3 0 1 -1.4 1 -9.5 5 | 6 | 6 |

**5. Текст программы:**

#include <iostream>

#include <fstream>

#include <iomanip>

#include <windows.h>

using namespace std ;

int main ( )

{ setlocale (0,""); // Подключение русского языка

short i; // описание переменных

float x0 , x1,x2;

ifstream infile ("in\_data.txt"); //данные берутся из файла

ofstream outfile ("out\_data.txt"); //данные записываются в файл

if (!infile) cout << "Входной файл не открыт!" << endl;

else {

//проверка первого

if(!(infile >> x0) ) cout << "Файл пуст!" << endl;

else{

//проверка второго элемента

if(!(infile >> x1)) {i=1; cout <<"Число локальных max последовательности = " << 1<<endl;

cout << "Локальный max последовательности = " << x0 << endl;

outfile <<"Число локальных max последовательности ="<<1<<endl;

}

else

{

if ( x0 > x1) i=1;

else i=0;

//проверка третьего элемента

if(!(infile >> x2))

{

if (i==1) cout << "Локальный max последовательности = " << x0 << endl;

else cout << "Локальный max последовательности = " << x1 << endl;

cout << "Число локальных max последовательности = " << 1 << endl;

outfile <<"Число локальных max последовательности = " <<1<<endl;

}

else

{

// продолжение поиска локальных максимумов

do {

if (( x0 <x1 )&&(x1>x2)) i++;

cout << x0 << "<" << x1 << ">" << x2 << " max: " << i << endl;

outfile<<setw(5)<<x0<<"<"<<setw(5)<<x1<<">"<<setw(5)<<x2<<"max:"<<i<<endl;

x0=x1;

x1=x2;

}

while (infile >> x2);

// проверка является ли последний элемент локальным максимумом

if ( x1 > x0) i++;

outfile << "Число локальных max последовательности = " << i << endl;

cout << "Число локальных max последовательности = " << i << endl;

}

}

}

infile.close();

outfile.close();

return 0;

}

}

**6. Вывод**

В ходе данной работы были получены практические навыки работы с индуктивными функциями на последовательностях, с задачами обработки последовательностей однотипных элементов с применением однопроходных алгоритмов с помощью языка программирования С++.