Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики і обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №1

З алгоритмів та методів обчислень

*Виконав:*

Студент групи ІО-12

Нестерук Ю.О.

*Перевірив:*

Подрубайло О.О.

м. Київ

2012 р.

**1. Тема завдання:**

Відповідно до варіанту завдання розробити блок-схеми обчислення виразів для лінійного алгоритму, алгоритму, що розгалужується та циклічного алгоритму. У відповідності до блок-схеми створити програму обчислення виразу на алгоритмічній мові Pascal.

**2. Завдання:**

|  |  |  |
| --- | --- | --- |
| Лінійний | Що розгалужується | Циклічний |
| C4=lg(a+b)-ln(a-b) | Обчислити вираз  для додатних значень показника степеня знаменника. | Обчислити середнє квадратичне для 100 значень змінної df |

**3. Блок-схеми алгоритмів:**



3.1 Лінійний

3.2 Що розгалужується



3.3 Циклічний

**4. Лістинг програми:**

***Клас MyReader***

**package** amo.reader;

**import** java.io.BufferedReader;

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.io.FileReader;

**import** java.io.IOException;

**import** java.io.InputStreamReader;

**public** **class** MyReader {

**private** **static** Mode *mode* = Mode.*Console*;

**private** **static** **boolean** *correct* = **false**;

**private** **static** BufferedReader *buffReader*;

**private** **static** File *f*;

**public** **static** **void** setMode(Mode m) {

*mode* = m;

}

**public** **static** **void** setFile(String fileName) {

*f* = **new** File(fileName);

}

**public** **static** **double**[] read(**double**[] darr, String[] sarr, CheckType[] cht) {

**if** (*mode* == Mode.*Console*) {

*buffReader* = **new** BufferedReader(**new** InputStreamReader(System.*in*));

**for**(**int** i = 0; i < darr.length; i++) {

**do** {

System.*out*.print("Введіть " + sarr[i] + " = ");

**try** {

darr[i] = Double.*valueOf*(*buffReader*.readLine());

*correct* = *check*(darr[i], cht[i]);

} **catch** (Exception e) {

System.*out*.println("Невірно введені дані.");

*correct* = **false**;

}

} **while**(!*correct*);

}

}

**else** {

**try** {

*buffReader* = **new** BufferedReader(**new** FileReader(*f*));

} **catch** (FileNotFoundException e1) {

e1.printStackTrace();

}

lb1: **for**(**int** i = 0; i < darr.length; i++) {

**do** {

**try** {

darr[i] = Double.*valueOf*(*buffReader*.readLine());

System.*out*.println("Зчитане значення " + sarr[i] + " = " + darr[i]);

*correct* = *check*(darr[i], cht[i]);

} **catch** (Exception e) {

System.*out*.println("Зчитані дані невірні.");

*correct* = **false**;

**break** lb1;

}

} **while**(!*correct*);

}

**try** {

*buffReader*.close();

} **catch** (IOException e) {

e.printStackTrace();

}

}

**return** darr;

}

**protected** **static** **boolean** check(**double** a, CheckType cht) {

**if**(cht == CheckType.*None*)

**return** **true**;

**if**(cht == CheckType.*NotZero*) {

**if**(a != 0 ) **return** **true**;

**else** System.*out*.println("Параметр повинен не дорівнювати нулю.");

}

**if**(cht == CheckType.*GreaterZero*) {

**if**(a > 0.0) **return** **true**;

**else** System.*out*.println("Параметр повинен бути більший нуля.");

}

**if**(cht == CheckType.*GreaterOrEqualsZero*) {

**if**(a >= 0.0) **return** **true**;

**else** System.*out*.println("Параметр повинен бути більший нуля або дорівнювати нулю.");

}

**if**(cht == CheckType.*LessZero*) {

**if**(a < 0.0) **return** **true**;

**else** System.*out*.println("Параметр повинен бути менший нуля.");

}

**if**(cht == CheckType.*LessOrEqualsZero*) {

**if**(a <= 0.0) **return** **true**;

**else** System.*out*.println("Параметр повинен бути менший нуля або дорівнювати нулю.");

}

**return** **false**;

}

}

***Перелічення CheckType***

**package** amo.reader;

**public** **enum** CheckType {

*None*, *NotZero*, *GreaterZero*, *GreaterOrEqualsZero*, *LessZero*, *LessOrEqualsZero*;

}

***Перелічення CheckType***

**package** amo.reader;

**public** **enum** Mode {

*Console*, *File*;

}

***Клас Linear***

**package** amo.lab1;

**import** amo.reader.\*;

**public** **class** Linear {

**public** **static** **double** *a*, *b*;

**public** **static** **double**[] *param*;

**public** **static** **void** main(String[] args) {

*param* = **new** **double**[2];

String[] paramNames = **new** String[] {"a", "b"};

CheckType[] cht = **new** CheckType[] {CheckType.*None*, CheckType.*None*};

//Режим читання з консолі

System.*out*.println("### Консольний режим");

**do** {

MyReader.*read*(*param*, paramNames, cht);

*a* = *param*[0];

*b* = *param*[1];

**if**(((*a*+*b*) > 0) && ((*a*-*b*) > 0))

**break**;

**else**

System.*out*.println("!!! Повинні виконуватися умови ((a+b) > 0) i ((a-b) > 0).\nПовторне введення даних.");

} **while**(**true**);

System.*out*.println("Введені дані:");

System.*out*.println("\ta = " + *a*);

System.*out*.println("\tb = " + *b*);

System.*out*.println("C4 = " + (Math.*log10*(*a*+*b*) - Math.*log10*(*a*-*b*)) + "\n");

//Режим читання з файлу

System.*out*.println("### Режим читання з файлу");

MyReader.*setMode*(Mode.*File*);

MyReader.*setFile*("1.txt");

MyReader.*read*(*param*, **new** String[] {"a", "b"}, **new** CheckType[] {CheckType.*None*, CheckType.*None*});

*a* = *param*[0];

*b* = *param*[1];

**if**(((*a*+*b*) > 0) && ((*a*-*b*) > 0)) {

System.*out*.println("Зчитані дані:");

System.*out*.println("\ta = " + *a*);

System.*out*.println("\tb = " + *b*);

System.*out*.println("C4 = " + (Math.*log10*(*a*+*b*) - Math.*log10*(*a*-*b*)) + "\n");

}

**else**

System.*out*.println("Повинні виконуватися умови ((a+b) > 0) i ((a-b) > 0).");

}

}

***Клас Branches***

**package** amo.lab1;

**import** amo.reader.CheckType;

**import** amo.reader.Mode;

**import** amo.reader.MyReader;

**public** **class** Branches {

**public** **static** **boolean** *correct*;

**public** **static** **double** *a*;

**public** **static** **double** *b*, *g*, *h*, *j*, *k*, *n*;

**public** **static** **double**[] *param*;

**public** **static** String[] *paramNames*;

**public** **static** CheckType[] *cht*;

**public** **static** **void** main(String[] args) {

*param* = **new** **double**[6];

*paramNames* = **new** String[] {"b", "g", "h", "j", "k", "n"};

*cht* = **new** CheckType[] {CheckType.*NotZero*, CheckType.*NotZero*, CheckType.*None*,

CheckType.*None*, CheckType.*None*, CheckType.*NotZero*};

//Режим читання з консолі

System.*out*.println("### Консольний режим");

**do** {

*correct* = *reading*();

**if**(!*correct*)

System.*out*.println("Повторне введення даних.");

} **while**(!*correct*);

*showInput*();

*a* = Math.*pow*(*g*, *g*\**h*)/(*n* \* Math.*pow*(*b*, *k* \* *j*));

System.*out*.println("a = " + *a* + "\n");

//Режим читання з файлу

System.*out*.println("### Режим читання з файлу");

MyReader.*setMode*(Mode.*File*);

MyReader.*setFile*("2.txt");

**if**(*reading*()) {

*showInput*();

*a* = Math.*pow*(*g*, *g*\**h*)/(*n* \* Math.*pow*(*b*, *k* \* *j*));

System.*out*.println("a = " + *a* + "\n");

}

**else**

System.*out*.println("Зчитування з файлу не вдалося.");

}

**public** **static** **boolean** reading() {

MyReader.*read*(*param*, *paramNames*, *cht*);

*b* = *param*[0];

*g* = *param*[1];

*h* = *param*[2];

*j* = *param*[3];

*k* = *param*[4];

*n* = *param*[5];

**if**((*k* \* *j*) > 0)

**return** **true**;

**else**

System.*out*.println("!!! Повинна виконуватися умова (k \* j) > 0.");

**return** **false**;

}

**public** **static** **void** showInput() {

System.*out*.println("\nВведені дані:");

System.*out*.println("\tb = " + *b*);

System.*out*.println("\tg = " + *g*);

System.*out*.println("\th = " + *h*);

System.*out*.println("\tj = " + *j*);

System.*out*.println("\tk = " + *k*);

System.*out*.println("\tn = " + *n*);

}

}

***Клас Cicles***

**package** amo.lab1;

**import** java.util.Random;

**public** **class** Cicle {

**public** **static** **int** *n* = 100;

**public** **static** **double** *df*, *sum*;

**public** **static** Random *r*;

**public** **static** **void** main(String[] args) {

*r* = **new** Random();

**for**(**int** i = 0; i < *n*; i++) {

*df* = *r*.nextInt(*n* + 1);

*sum* += *df*\**df*;

}

*sum* = Math.*sqrt*(*sum*/*n*);

System.*out*.println("Середнє квадратичне значення змінної df = " + *sum*);

}

}

**5. Роздруківка результатів виконання програми:**

**5.1 Лінійний:**

### Консольний режим

Введіть a = 5

Введіть b = 3

Введені дані:

a = 5.0

b = 3.0

C4 = 0.6020599913279623

### Режим читання з файлу

Зчитане значення a = 9.0

Зчитане значення b = 4.0

Зчитані дані:

a = 9.0

b = 4.0

C4 = 0.41497334797081786

**5.2 Що розгалужується:**

### Консольний режим

Введіть b = 3

Введіть g = 4

Введіть h = 1

Введіть j = -9

Введіть k = 5

Введіть n = 1

!!! Повинна виконуватися умова (k \* j) > 0.

Повторне введення даних.

Введіть b = 3

Введіть g = 4

Введіть h = 1

Введіть j = 9

Введіть k = 5

Введіть n = 1

Введені дані:

b = 3.0

g = 4.0

h = 1.0

j = 9.0

k = 5.0

n = 1.0

a = 8.665298004248187E-20

### Режим читання з файлу

Зчитане значення b = 1.0

Зчитане значення g = 2.0

Зчитане значення h = 3.0

Зчитане значення j = 4.0

Зчитане значення k = 5.0

Зчитане значення n = 6.0

Введені дані:

b = 1.0

g = 2.0

h = 3.0

j = 4.0

k = 5.0

n = 6.0

a = 10.666666666666666

**5.3 Циклічний:**

Середнє квадратичне значення змінної df = 59.042103621060114

**7. Аналіз результатів:**

Виконуючи цю лабораторну роботу, я навчився створювати блок-схеми лінійного, розгалуженого та циклічного алгоритмів за допомогою редактора схем та діаграм Microsoft Office Visio. У відповідності до блок-схеми я створив програми для обчислення виразів для даних алгоритмів.

Результати роботи програм демонструють їхню працездатність та можливість перевірки ними коректності введених даних. У разі неправильного формату введених даних або невиконанні певних умов (наприклад, заборони ділення на нуль) програмою видається повідомлення про помилку.