Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 4

тема «Линейные алгоритмы в Java»

по дисциплине «Информатика»

Выполнил: студент группы ИСТ-21-1б Носов Д.О.

Проверил: ассистент каф. ВММБ Нетбай Г.В.

Пермь, 2022

**Содержание**

[Задание 1 3](#_Toc117303857)

[1.1. Постановка задачи 3](#_Toc117303858)

[1.2. Решение задачи, код программы 3](#_Toc117303859)

[1.3.1. Тестирование работы программы с проверкой 3](#_Toc117303860)

[1.3.2. Тестирование работы программы с проверкой 4](#_Toc117303861)

[Задание 2 6](#_Toc117303862)

[2.1. Постановка задачи 6](#_Toc117303863)

[2.2. Решение задачи, код программы 6](#_Toc117303864)

[2.3. Тестирование работы программы с проверкой 6](#_Toc117303865)

[Задание 3 8](#_Toc117303866)

[3.1. Постановка задачи 8](#_Toc117303867)

[3.2. Решение задачи, код программы 8](#_Toc117303868)

[3.3. Тестирование работы программы с проверкой 8](#_Toc117303869)

[Задание 4 10](#_Toc117303870)

[4.1. Постановка задачи 10](#_Toc117303871)

[4.2. Решение задачи, код программы 10](#_Toc117303872)

[4.3. Тестирование работы программы с проверкой 10](#_Toc117303873)

[Задание 5 12](#_Toc117303874)

[5.1. Постановка задачи 12](#_Toc117303875)

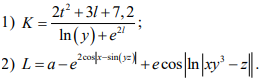
[5.2. Решение задачи, код программы 12](#_Toc117303876)

[5.3. Тестирование работы программы с проверкой 12](#_Toc117303877)

# Задание 1

## 1.1. Постановка задачи

Вычислить значения двух функций, вычисление значения функций организовать в виде отдельных методов класса:



## 1.2. Решение задачи, код программы

import static java.lang.Math.\*;  
public class zad1 {  
 public static String K(double t, double l, double y) {  
 if (y <= 0 || *log*(y) + *exp*(2\*l) == 0) {  
 return "Ошибка";  
 } else {  
 double Chislitel = 2 \* *pow*(t, 2) + 3 \* l + 7.2;  
 double Znamenatel = *log*(y) + *exp*(2\*l);  
 return String.*valueOf*(Chislitel / Znamenatel);  
 }  
 }  
 public static String L(double a, double x, double y, double z) {  
 if (y \* z < -1 || y \* z > 1 || x - *sin*(y \* z) < -1 || x - *sin*(y \* z) > 1 || *abs*(x \* *pow*(y, 3) - z) == 0 || *abs*(*log*(*abs*(x \* *pow*(y, 3) - z))) > 1) {  
 return "Ошибка";  
 } else {  
 double first = a - *pow*(*E*, 2 \* *abs*(*cos*(x - *sin*(y \* z))));  
 double sec = *E* \* *cos*(*abs*(*log*(*abs*(x \* *pow*(y, 3) - z))));  
 return String.*valueOf*(first + sec);  
 }  
 }  
 public static void main(String[] args) {  
 String K = *K*(0,1,-80);  
 String L = *L*(0,1,3,4);  
 if (K.equals("Ошибка")) {  
 System.*out*.println("Ошибка в данных 1-го уравнения");  
 } else {  
 System.*out*.println("K = " + K);  
 }  
 if (L.equals("Ошибка")) {  
 System.*out*.println("Ошибка в данных 2-го уравнения");  
 } else {  
 System.*out*.println("L = " + L);  
 }  
 }

## 1.3.1. Тестирование работы программы с проверкой

Для проверки задачи в MS Excel создана таблица данных в которой в ячейку А2 записана переменная t, в ячейку B2 – l и в ячейку C2 – y. В ячейку D2 записана формула для вычисления значения функции K.

Формула для вычисления функции K:

D2) =ЕСЛИ(C2<=0; "Ошибка данных"; ЕСЛИ(LN(C2) + EXP(2\*B2)=0; "Ошибка данных"; (2\*A2^2+3\*B2+7,2) / (LN(C2) + EXP(2\*B2)))).

На рис. 1.1 представлен вид решения в MS Excel.

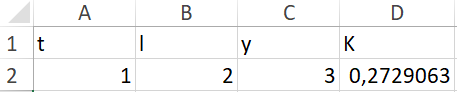


Рис. 1.1 Решение задачи в MS Excel

Далее в таблице 1.1 представлено тестирование работы программы с проверкой решения задачи на языке Java с решением задачи в MS Excel.

Таблица 1.1

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Решение MS Excel |
| 1 |  |  |
| 2 |  |  |

Сравнение решения задачи с использованием двух прикладных пакетов показала, что решения задачи в Java и MS Excel совпадает. Данный факт подтверждает правильность написанного кода программы. Неопределенностей при решении задачи выявлено не было, возможно данная ситуация связанна с малым количеством проверок данных.

## 1.3.2. Тестирование работы программы с проверкой

Для проверки задачи в MS Excel создана таблица данных в которой в ячейку А2 записана переменная a, в ячейку B2 – x, в ячейку C2 – y и в ячейку D2 - z. В ячейку E2 записана формула для вычисления значения функции L.

Формула для вычисления функции L:

E2) =ЕСЛИ(ИЛИ(C2\*D2 < -1;C2\*D2 > 1; ABS(B2\*C2^3 - D2) = 0); "Ошибка данных"; ЕСЛИ(ИЛИ(ABS(B2 - SIN(C2\*D2)) > 1;ABS(LN(ABS(B2\*C2^3 - D2))) > 1);"Ошибка данных"; A2-EXP(2\*COS(ABS(B2 - SIN(C2\*D2))) )+ EXP(1)\*COS(ABS(LN(ABS(B2\*C2^3 - D2)))))).

На рис. 1.2 представлен вид решения в MS Excel.

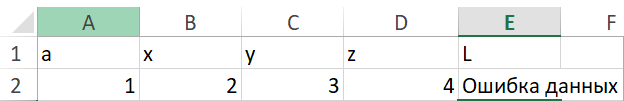


Рис. 1.2 Решение задачи в MS Excel

Далее в таблице 1.2 представлено тестирование работы программы с проверкой решения задачи на языке Java с решением задачи в MS Excel.

Таблица 1.2

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Решение MS Excel |
| 1 |  |  |
| 2 |  |  |

Сравнение решения задачи с использованием двух прикладных пакетов показала, что решения задачи в Java и MS Excel совпадает. Данный факт подтверждает правильность написанного кода программы. Неопределенностей при решении задачи выявлено не было, возможно данная ситуация связанна с малым количеством проверок данных.

# Задание 2

## 2.1. Постановка задачи

Создать программу для решения задачи по физике, для произвольно введенных значений параметров. Расстояние между пристанями A и B равно S км. Из A в B по течению реки отправился плот, а через час вслед за ним отправилась яхта, которая, прибыв в пункт B, тотчас повернула обратно и возвратилась в A. К этому времени плот прошел L км. Найдите скорость яхты в неподвижной воде, если скорость течения реки равна v км/ч

## 2.2. Решение задачи, код программы

import java.util.Scanner;  
  
public class zad2 {  
 public static void main(String[] args) {  
 Scanner scn = new Scanner(System.*in*);  
 System.*out*.println("Введите данные S, v, L:");  
 double S, v, L, b, D, x1, x2;  
 S = scn.nextDouble();  
 v = scn.nextDouble();  
 L = scn.nextDouble();  
 /\* S(1/(x+v) + 1/(x-v)) = L/v - 1 пиводим в  
 x^2 - 2Svx/(L-v) - v^2 = 0\*/  
 if (L – v != 0) {  
 b = -2 \* S \* v / (L - v);  
 D = Math.*pow*(b, 2) + 4 \* v \* v;  
 x1 = (-b + Math.*pow*(D, 1 / 2f)) / 2;  
 x2 = (-b - Math.*pow*(D, 1 / 2f)) / 2;  
 if (x1 > 0) {  
 System.*out*.println("Корень уравнения " + x1);  
 } else if (x2 > 0) {  
 System.*out*.println("Корень уравнения " + x2);  
 } else {  
 System.*out*.println("Нет решений");  
 }  
 } else {  
 System.*out*.println("Нет решений");  
 }  
 }  
}

## 2.3. Тестирование работы программы с проверкой

Для проверки задачи в MS Excel создана таблица данных в которой в ячейку А2 записана переменная S, в ячейку B2 – v и в ячейку C2 – L. В ячейки D2 и E2 записаны формулы для вычисления значения корней x1 и x2, а в ячейке B3 ответ с учетом ОДЗ.

Формулы для вычисления корней x1, x2 и ответа:

D2) =ЕСЛИ(C2=B2; 0;(2\*A2\*B2/(C2-B2) + ((-2\*A2\*B2/(C2-B2))^2 + 4\*B2^2)^(1/2))/2);

E2) =ЕСЛИ(C2=B2; 0;(2\*A2\*B2/(C2-B2) - ((-2\*A2\*B2/(C2-B2))^2 + 4\*B2^2)^(1/2))/2);

B3) =ЕСЛИ(И(D2>0;E2>0); "x1 и x2 корни"; ЕСЛИ(D2>0;"x1 корень"; ЕСЛИ(E2>0;"x2 корень"; "Нет корней"))).

На рис. 2 представлен вид решения в MS Excel.

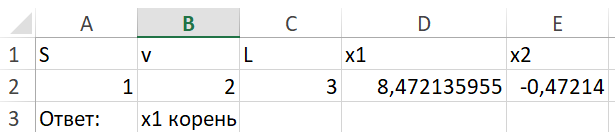


Рис. 2. Решение задачи в MS Excel

Далее в таблице 2 представлено тестирование работы программы с проверкой решения задачи на языке Java с решением задачи в MS Excel.

Таблица 2

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Решение MS Excel |
| 1 |  |  |
| 2 |  |  |

Сравнение решения задачи с использованием двух прикладных пакетов показала, что решения задачи в Java и MS Excel совпадает. Данный факт подтверждает правильность написанного кода программы. Неопределенностей при решении задачи выявлено не было, возможно данная ситуация связанна с малым количеством проверок данных.

# Задание 3

## 3.1. Постановка задачи

Создать программу для решения задачи по геометрии. Определить углы параллелограмма, если даны две его высоты h1, h2 и периметр p.

## 3.2. Решение задачи, код программы

import java.util.Scanner;  
  
public class zad3 {  
 public static void main(String[] args) {  
 Scanner scn = new Scanner(System.*in*);  
 System.*out*.println("Введите данные p, h1, h2:");  
 double p = scn.nextDouble();  
 double h1 = scn.nextDouble();  
 double h2 = scn.nextDouble();  
 if (((2 \* (h1 + h2) / p) < -1) || ((2 \* (h1 + h2) / p) > 1)) {  
 System.*out*.println("Ошибка данных");  
 } else {  
 double Ugol1 = Math.*asin*(2 \* (h1 + h2) / p) \*180/Math.*PI*;  
 double Ugol2 = 180 - Ugol1;  
 System.*out*.println("Меньший угол равен " + Ugol1);  
 System.*out*.println("Больший угол равен " + Ugol2);  
 }  
 }  
}

}

## 3.3. Тестирование работы программы с проверкой

Для проверки задачи в MS Excel создана таблица данных в которой в ячейку А2 записана переменная p, в ячейку B2 – h1 и в ячейку C2 – h2. В ячейки D2 и E2 записаны формулы для вычисления значения функций Ugol1 и Ugol2.

Формулы для вычисления функций Ugol1 и Ugol2:

D2) =ЕСЛИ(ИЛИ((2\*(B2+C2) / A2) < -1;(2\*(B2+C2) / A2) > 1);"Ошибка данных"; ASIN(2\*(B2+C2) / A2)\*180/ПИ());

E2) =ЕСЛИ(ИЛИ((2\*(B2+C2) / A2) < -1;(2\*(B2+C2) / A2) > 1);"Ошибка данных"; 180 - D2).

На рис. 3 представлен вид решения в MS Excel.

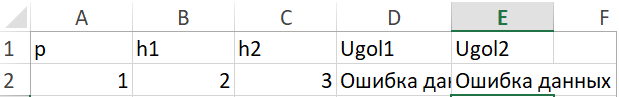


Рис. 3. Решение задачи в MS Excel

Далее в таблице 3 представлено тестирование работы программы с проверкой решения задачи на языке Java с решением задачи в MS Excel.

Таблица 3

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Решение MS Excel |
| 1 |  |  |
| 2 |  |  |

Сравнение решения задачи с использованием двух прикладных пакетов показала, что решения задачи в Java и MS Excel совпадает. Данный факт подтверждает правильность написанного кода программы. Неопределенностей при решении задачи выявлено не было, возможно данная ситуация связанна с малым количеством проверок данных.

# Задание 4

## 4.1. Постановка задачи

Создать программу. Столовая израсходовала за N месяца(ев) X кг овощей: в первый месяц Y этих овощей, во второй месяц – в K раза меньше, чем в первый, а остальные месяцы овощи расходовались поровну. По сколько килограммов овощей расходовала столовая в последний месяц?

## 4.2. Решение задачи, код программы

import java.util.Scanner;  
  
public class zad4 {  
 public static void main(String[] args) {  
 Scanner scn = new Scanner(System.*in*);  
 System.*out*.println("Введите данные N, X, Y, K:");  
 double N = scn.nextDouble();  
 double X = scn.nextDouble();  
 double Y = scn.nextDouble();  
 double K = scn.nextDouble();  
 double LastMes = (X - 2 \* Y/K) / (N - 2);  
 if (LastMes > 0 & (N != 2 || K != 0)) {  
 System.*out*.println("Кол-во овощей израсходованных в последний месяц: " + LastMes + " Кг");  
 } else {  
 System.*out*.println("Ошибка данных");  
 }  
 }  
}

## 4.3. Тестирование работы программы с проверкой

Для проверки задачи в MS Excel создана таблица данных в которой в ячейку А2 записана переменная N, в ячейку B2 – X, в ячейку C2 – Y и в ячейку D2 – K. В ячейке E2 записана формула для вычисления значения Y в конце месяца - LastMes.

Формулы для вычисления функций LastMes:

E2) =ЕСЛИ(ИЛИ(A2 <> 2;D2 <> 0); ЕСЛИ((B2-2\*C2/D2)/(A2-2) > 0; (B2-2\*C2/D2)/(A2-2); "Ошибка данных");"Ошибка данных").

На рис. 4 представлен вид решения в MS Excel.

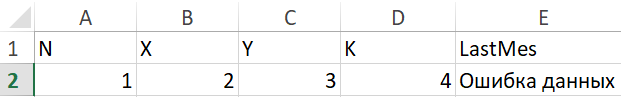


Рис. 4. Решение задачи в MS Excel

Далее в таблице 4 представлено тестирование работы программы с проверкой решения задачи на языке Java с решением задачи в MS Excel.

Таблица 4

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Решение MS Excel |
| 1 |  |  |
| 2 |  |  |

Сравнение решения задачи с использованием двух прикладных пакетов показала, что решения задачи в Java и MS Excel совпадает. Данный факт подтверждает правильность написанного кода программы. Неопределенностей при решении задачи выявлено не было, возможно данная ситуация связанна с малым количеством проверок данных.

# Задание 5

## 5.1. Постановка задачи

Создать метод (вне метода main), который вычисляет проекцию произвольной точки на прямую проходящую чрез две точки. Пользователь вводит через консоль координаты точек, через которые проходит прямая, координаты произвольной точки и получает в ответ проекцию точки, на заданную прямую.

## 5.2. Решение задачи, код программы

import java.util.Scanner;  
  
public class zad5 {  
 public static void main(String[] args) {  
 Scanner scn = new Scanner(System.*in*);  
 System.*out*.println("Введите данные первой точки на прямой: x, y:");  
 double x1 = scn.nextDouble();  
 double y1 = scn.nextDouble();  
 System.*out*.println("Введите данные второй точки на прямой: x, y:");  
 double x2 = scn.nextDouble();  
 double y2 = scn.nextDouble();  
 while (x2 == x1 || y2 == y1){  
 System.*out*.println("Первая и вторая точки совпали!");  
 System.*out*.println("Введите данные второй точки на прямой: x, y:");  
 x2 = scn.nextDouble();  
 y2 = scn.nextDouble();  
 }  
 System.*out*.println("Введите данные проецируемой точки: x, y:");  
 double x3 = scn.nextDouble();  
 double y3 = scn.nextDouble();  
 double[] xy = *ProgectXY*(x1,y1,x2,y2,x3,y3);  
 System.*out*.println("Координаты проекции: x = " + xy[0] + " и y = " + xy[1]);  
 }  
 /\* (x - x1)/(x2 - x1) = (y - y1)/(y2 - y1) = t - канон. уравнение прямой  
 выводим x = (x2 - x1)t + x1 и y = (y2 - y1)t + y1  
 из (x2 - x1)x + (y2 - y1)y - (x2 - x1)x3 - (y2 - y1)y3 = 0 получаем t \*/  
 public static double[] ProgectXY(double x1, double y1, double x2,  
 double y2, double x3, double y3){  
 double[] xy = new double[2];  
 double m = x2 - x1;  
 double p = y2 - y1;  
 double t = (m\*x3 + p\*y3 - m\*x1 - p\*y1) / (m\*m + p\*p);  
 xy[0] = m\*t + x1; /\* это x \*/  
 xy[1] = p\*t + y1; /\* а это y \*/  
 return xy;  
 }  
}

## 5.3. Тестирование работы программы с проверкой

Для проверки задачи в MS Excel создана таблица данных в которой в ячейку B1 записана переменная x1, в ячейку B2 – y1, в ячейку B3 – x2, в ячейку B4 – y2, в ячейку B5 – x3 и в ячейку B6 – y3. В ячейки E1, E2, E3, E4 и E5 записаны формулы для вычисления координат направляющего вектора – m и p, коэффициент для расчёта параметрического уравнения прямой – t, проекции координат - x3 и y3 на прямую проведенную через точки с координатами (x2;y2), (x1,y1).

Формулы для вычисления функций a и b:

E1) =ЕСЛИ(B3 = B1; "Введите различные точки x2 и x1!"; B3-B1);

E2) =ЕСЛИ(B4=B2; "Введите различные точки y2 и y1!"; B4-B2);

E3) =(E1\*B5+E2\*B6-E1\*B1-E2\*B2)/(E1^2+E2^2);

E4) =E1\*E3+B1;

E5) =E2\*E3+B2.

На рис. 5 представлен вид решения в MS Excel.

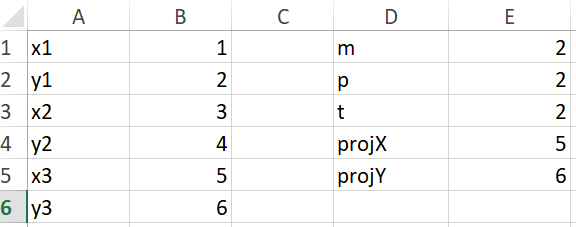


Рис. 5. Решение задачи в MS Excel

Далее в таблице 5 представлено тестирование работы программы с проверкой решения задачи на языке Java с решением задачи в MS Excel.

Таблица 5

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Решение MS Excel |
| 1 |  |  |
| 2 |  |  |

Сравнение решения задачи с использованием двух прикладных пакетов показала, что решения задачи в Java и MS Excel совпадает. Данный факт подтверждает правильность написанного кода программы. Неопределенностей при решении задачи выявлено не было, возможно данная ситуация связанна с малым количеством проверок данных.