Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 4

Вариант 1

тема «Линейные алгоритмы в Java»

по дисциплине «Информатика»

Выполнил: студент группу ИСТ-22-1б Ипанов М.В.

Проверил: доцент каф. ВММБ Каменских А.А.

Пермь, 2021

**Содержание**

[Задание 1 3](#_Toc118064081)

[1.1. Постановка задачи 3](#_Toc118064082)

[1.2. Решение задачи, код программы 3](#_Toc118064083)

[1.3. Тестирование работы программы с проверкой 3](#_Toc118064084)

[Задание 2 5](#_Toc118064085)

[2.1. Постановка задачи 5](#_Toc118064086)

[2.2. Решение задачи. Код программы 5](#_Toc118064087)

[2.3. Тестирование работы программы 5](#_Toc118064088)

[Задание 3 7](#_Toc118064089)

[3.1. Постановка задачи 7](#_Toc118064090)

[3.2. Решение задачи. Код программы 7](#_Toc118064091)

[3.3 Проверка решения 7](#_Toc118064092)

[Задание 4 9](#_Toc118064093)

[4.1. Постановка задачи 9](#_Toc118064094)

[4.2. Решение задачи. Код программы 9](#_Toc118064095)

[4.3. Проверка решения 10](#_Toc118064096)

[Задание 5 11](#_Toc118064097)

[5.1. Постановка задачи 11](#_Toc118064098)

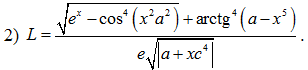
[5.2. Решение задачи. Код программы 12](#_Toc118064099)

[5.3. Проверка решения 13](#_Toc118064100)

# Задание 1

## 1.1. Постановка задачи

Даны произвольные **y, f, x, a, c**. Вычислить **G, L**, если:

## 1.2. Решение задачи, код программы

package com.company;  
import java.util.\*;  
import static java.lang.Math.\*;  
class Main {  
 public static void main(String args[]) {  
 Scanner in = new Scanner(System.*in*);  
   
 double y, f, G;  
 System.*out*.println("Введите переменную y");  
 y = in.nextDouble();  
 System.*out*.println("Введите переменную f");  
 f = in.nextDouble();  
  
 G = (*exp*(2\*y)+*sin*(f))/(*log*(3.8\*y+f));  
 System.*out*.printf("G = %.6f", G);  
 System.*out*.println();  
   
 double x, a, c, L, L1, L2, L3;  
 System.*out*.println("Введите переменную x");  
 x = in.nextDouble();  
 System.*out*.println("Введите переменную a");  
 a = in.nextDouble();  
 System.*out*.println("Введите переменную c");  
 c = in.nextDouble();  
  
 L = (*sqrt*(*exp*(x)-*pow*(*cos*(*pow*(x, 2)\**pow*(a, 2)),4))+*pow*(*atan*(a-*pow*(x,5)),4)) /  
 (Math.*E*\**sqrt*(*abs*(a+x\**pow*(c, 4))));  
 System.*out*.printf("L = %.6f", L);  
 }  
}

## 1.3. Тестирование работы программы с проверкой

Для проверки задачи в MS Excel созданы 2 таблицы данных, где для вычисления функции С в ячейку С2 была записана функция (1), в A2 переменная y, в ячейку B2 – f, а для вычисления функции L в ячейку В5 была записана функция (2) а в ячейки A5, B5, C5 записаны переменные x, a, c соответственно.

Формулы для вычисления функций a и b:

1) = (EXP(2\*A2)+SIN(B2))/LN(3,8\*A2+B2)

2) = (КОРЕНЬ(EXP(A5)-COS(A5^2\*B5^2)^4)+ATAN(B5-A5^5)^4)/

(EXP(1)\*КОРЕНЬ(ABS(B5+A5\*C5^4)))

На рис. 1 представлен вид решения в MS Excel.

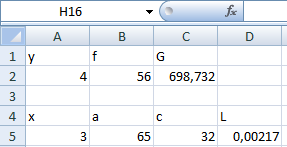


Рис. 1. Решение задачи в MS Excel

Далее в таблице 1 представлено тестирование работы программы с проверкой решения задачи на языке Java с решением задачи в MS Excel.

Таблица 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. При вводе некоторых данных программа выводила NaN – неопределённость или Inf – бесконечность, при анализе этих результатов вручную выяснилось, что действительно эти примеры или не имели решения (например sin(23)) или получались на столько большие числа (exp(12345)), что не хватало размерности переменных, чтобы программа смогла посчитать эти числа.

# Задание 2

## 2.1. Постановка задачи

Создать программу для решения задачи по физике, для произвольно введенных значений параметров. Между двумя пунктами, расположенными на реке на расстоянии  км один от другого, курсирует катер, который, идя по течению, проходит это расстояние за время  ч, а против течения, – за время  ч. Определить скорость течения реки  и скорость катера  относительно воды.

## 2.2. Решение задачи. Код программы

package com.company;

import java.util.\*;

import static java.lang.Math.\*;

class Main {

public static void main(String args[]) {

Scanner in = new Scanner(System.in);

double S, t\_1, t\_2, U\_k, U\_r;

System.out.print("Введите расстояние между пунктами: ");

S = in.nextDouble();

System.out.print("Ввидете время по течению: ");

t\_1 = in.nextDouble();

System.out.print("Ввидете время против течения: ");

t\_2 = in.nextDouble();

if (S >= 0 && t\_1 > 0 && t\_2 > 0 && t\_1 <= t\_2){

U\_k = S / (2 \* t\_1) + S / (2 \* t\_2);

U\_r = S / (2 \* t\_1) - S / (2 \* t\_2);

System.out.printf("Скорость катера = %.1f", U\_k);

System.out.println();

System.out.printf("Скорость реки = %.1f", U\_r);

}

else

System.out.print("Проверьте пункты:\n" +

" 1 - Время и расстояние не могут быть отрицательными,\n" +

" 2 - Время не может равнятся 0,\n" +

" 3 - Скорость по течению должна быть больше скорости против течения.");

}

}

## 2.3. Тестирование работы программы

Для проверки решения задача с некоторыми входными данными была прорешена вручную и составлена таблица соответствия решения в программе:

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Ответы полученные вручную |
| 1 |  | U1 = 5, U2 = 1 |
| 2 |  | U1 = 252, U2 = 248 |
| 3 |  | Данные не соответствуют действительности |

В ходе проверки выяснилось, что программы выполняет все свои функции и также отсеивает некорректные данные, которые может ввести пользователь.

# Задание 3

## 3.1. Постановка задачи

Создать программу для решения задачи по геометрии. Заданы *R*, *r*, *h* – измерения усеченного конуса. Вычислить площадь поверхности и объем усеченного конуса.

## 3.2. Решение задачи. Код программы

kage com.company;

import java.util.\*;

import static java.lang.Math.\*;

class Main {

public static void main(String args[]) {

Scanner in = new Scanner(System.in);

double R, r, h, S\_pp, V;

System.out.print("Введите измерения конуса:\n - радиус большего основания = ");

R = in.nextDouble();

System.out.print(" - радиус меньшего основания = ");

r = in.nextDouble();

System.out.print(" - высота конуса = ");

h = in.nextDouble();

if (R >= 0 && r >= 0 && h >= 0){

S\_pp = Math.PI \* ( (r+R) \* sqrt(pow(h,2)+pow(abs(R-r), 2) ) +

pow(r, 2) + pow(R, 2));

V = Math.PI \* h / 3 \* ( pow(r, 2) + pow(R, 2) + r\*R );

System.out.printf("S = %.1f; V = %.1f", S\_pp, V);

}

else

System.out.print("Некорректные данные, проверте следующие условия:\n" +

" 1 - Ни один из параметров не может быть отрицательным.");

}

## 3.3. Проверка решения

Для проверки решения была создана таблица Excel, где в ячейках A1, B1, D1 вносятся переменные r, R, h соответственно, а в ячейках 4A, 4B записаны формулы для вычисления S и V:

A4) =ПИ()\*((A2+B2)\*КОРЕНЬ(СТЕПЕНЬ(C2;2)+СТЕПЕНЬ(A2+B2;2))+  
СТЕПЕНЬ(A2;2)+СТЕПЕНЬ(B2;2))

B4) =ПИ()\*C2/3\*(СТЕПЕНЬ(A2;2)+A2\*B2+СТЕПЕНЬ(B2;2))

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Решение MS Excel |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |

Тестирование работы программы и проверка результатов решения

В ходе небольшого количества проверок никаких ошибок выявлено не выявлено, программа работала корректно со всеми контрольными данными.

# Задание 4

## 4.1. Постановка задачи

Создать программу для перевода одной величины в другие. У пользователя есть A руб. Выполнить перевод денег сначала в доллары, а потом в евро. Текущие курсы обмена валют пользователь вводит через консоль. Так же определить, сколько рублей пользователь потерял при конвертации валют.

## 4.2. Решение задачи. Код программы

import java.util.\*;

import static java.lang.Math.\*;

class Main {

public static void main(String args[]) {

Scanner in = new Scanner(System.in);

double D\_in, E\_in, DE\_in, input, D, E, DE, output;

System.out.print("Введите текущие курсы волют:\n (десятичную часть следует отделять запятой)\n - 1 Доллар = рублей ");

D\_in = in.nextDouble();

System.out.print(" - 1 Евро = рублей ");

E\_in = in.nextDouble();

System.out.print(" - 1 Евро = долларов ");

DE\_in = in.nextDouble();

System.out.print("Вы имеете рублей: ");

input = in.nextDouble();

System.out.print("Это:\n");

if (abs(D\_in) == D\_in && abs(E\_in) == E\_in && abs(DE\_in) == DE\_in && abs(input) == input) {

D = input/D\_in;

System.out.printf(" - В Долларах = %.2f\n", D);

E = input/E\_in;

System.out.printf(" - В Евро = %.2f\n", E);

DE = input - D / DE\_in \* E\_in;

System.out.printf(" - Потери при конвекции Рубли -> Доллары -> Евро -> Рубли = %.2f", DE);

if (abs(DE) != DE) {

System.out.print("Поздравляю, вы в плюсе!");

}

}

else

System.out.print("Некорректные данные, проверте следующие условия:\n" +

" 1 - Ни один из параметров не может быть отрицательным.");

}

}

## 4.3. Проверка решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Решение MS Excel |
| 1 | Текущий курс | |
|  |  |
| 2 |  |  |
| 3 |  |  |

При тестировании программы на некоторых реальных и произвольных данных при правильном вводе данных ошибок выявлено не было. Также программа сообщает если данные некорректны (с минусом).

PS: Я был удивлён, когда в ходе решения узнал, что на конвекции волют можно «уйти в плюс».

# 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.\*;

import static java.lang.Math.\*;

class Main {

public static void main(String args[]) {

Scanner in = new Scanner(System.in);

double[][] points = new double[3][3];

String[][] ABC\_xyz = {{"A", "B", "C"}, {"x", "y", "z"}};

for (int i = 0; i < 3; i++) {

if (i + 1 == 3) {

System.out.print("Введите координаты точки проэкцию которой нужно найти:\n");

} else {

System.out.print("Введите координаты " + (i+1) + " точки:\n");

}

for (int j = 0; j < 3; j++) {

System.out.print(ABC\_xyz[0][i] + "\_" + ABC\_xyz[1][j] + " = ");

points[i][j] = in.nextDouble();

}

}

System.out.println("Координаты искомой точки:");

double C[] = Projection\_point(points);

for (int i = 0; i < 3; i++) {

System.out.println(ABC\_xyz[1][i] + " = " + C[i]);

}

}

public static double[] Projection\_point(double[][] points) {

double lu, projection\_coordinates[] = new double[3];

lu = ( (points[2][0] - points[0][0]) \* (points[1][0] - points[0][0]) +

(points[2][1] - points[0][1]) \* (points[1][1] - points[0][1]) +

(points[2][2] - points[0][2]) \* (points[1][2] - points[0][2]) )

/

( pow(points[1][0] - points[0][0], 2) +

pow(points[1][1] - points[0][1], 2) +

pow(points[1][2] - points[0][2], 2) );

for (int i = 0; i < 3; i++)

projection\_coordinates[i] = points[0][i] + (points[1][i] - points[0][i]) \* lu;

return projection\_coordinates;

}

}

## 5.3. Проверка решения

Для проверки решения будем использовать уже решённые прототипы этих задач с интернет ресурсов:

* <https://matworld.ru/analytic-geometry/proekcija-tochki-na-prjamuju.php>
* https://zaochnik.com/spravochnik/matematika/prjamaja-ploskost/proektsija-tochki-na-prjamuju-koordinaty-proektsii/

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Прототип задания с ответом |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |

В ходе проверки программа работала корректно со всеми введёнными данными. Ошибок выявлено не было. Ввод и вывод данных вполне удобный в использовании, достаточно наглядный.

\*Конечно, лучше было бы сделать проверку решения в Excel, но… такие длинные формулы очень лень переписывать туда, так что проверил с помощью готовых решений)