Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 5

Вариант 17

тема «Ветвящиеся алгоритмы»

по дисциплине «Информатика»

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

Проверил: доцент каф. ВММБ Каменских А.А.

Пермь, 2022

**Содержание**

[Задание 1 3](#_Toc118672523)

[1.1. Постановка задачи 3](#_Toc118672524)

[1.2. Решение задачи, код программы 3](#_Toc118672525)

[1.3. Тестирование работы программы 4](#_Toc118672526)

[Задание 2 5](#_Toc118672527)

[2.1. Постановка задачи 5](#_Toc118672528)

[2.2. Решение задачи. Код программы 5](#_Toc118672529)

[2.3. Тестирование работы программы 6](#_Toc118672530)

[Задание 3 7](#_Toc118672531)

[3.1. Постановка задачи 7](#_Toc118672532)

[3.2. Решение задачи. Код программы 7](#_Toc118672533)

[3.3. Проверка решения 8](#_Toc118672534)

[Задание 4 9](#_Toc118672535)

[4.1. Постановка задачи 9](#_Toc118672536)

[4.2. Решение задачи. Код программы 9](#_Toc118672537)

[4.3. Проверка решения 10](#_Toc118672538)

[Задание 5 11](#_Toc118672539)

[5.1. Постановка задачи 11](#_Toc118672540)

[5.2. Решение задачи. Код программы 12](#_Toc118672541)

[5.3. Проверка решения 13](#_Toc118672542)

[Задание 6 14](#_Toc118672543)

[6.1. Постановка задачи 14](#_Toc118672544)

[6.2. Решение задачи. Код программы 15](#_Toc118672545)

[6.3. Проверка решения 17](#_Toc118672546)

[Задание 7 18](#_Toc118672547)

[7.1. Постановка задачи 18](#_Toc118672548)

[7.2. Решение задачи. Код программы 19](#_Toc118672549)

[7.3. Проверка решения 20](#_Toc118672550)

# Задание 1

## 1.1. Постановка задачи

Написать программу, выводящую номер этажа, где должен остановиться лифт, если известно, что в  этажном доме один подъезд, на этаже по 5 квартир. При входе в лифт вводится номер квартиры. В случае ошибочного ввода номера квартиры, программа должна сообщать об этом.

## 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 x, y;

int floor;  
 System.*out*.print("Сколько в доме этажей? ");  
 floor = in.nextInt();  
 System.*out*.print("Квартира ");  
 int i\_1 = in.nextInt();  
 if (*abs*(i\_1) == i\_1 && i\_1 <= 5 \* floor) {  
 System.*out*.print(i\_1 / 5 + 1);  
 } else {  
 System.*out*.print("error");  
 }

}}

## 1.3. Тестирование работы программы

Таблица 1

Тестирование работы программы и проверка результатов решения

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

Сравнение решения в Java и MS Excel показали правильность решения.  
Формула в ячейке вывода:  
=ЕСЛИ(И(C3 > 0; C3 <= 5\*C2); ЧАСТНОЕ(C3 - 1; 5) + 1; "error")

# Задание 2

## 2.1. Постановка задачи

Написать программу, которая находит наименьшее и наибольшее абсолютное значение среди трех заданных чисел.

## 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 x, y;

System.*out*.print("Введите 3 числа:\n");  
 int e\_high, e\_low;  
  
 double[] e\_1 = new double[3];  
 for (int i\_2 = 0; i\_2 < 3; i\_2++)  
 e\_1[i\_2] = in.nextDouble();  
  
 double[] e = new double[3];  
 for (int i\_2 = 0; i\_2 < 3; i\_2++)  
 e[i\_2] = *abs*(e\_1[i\_2]);  
  
 if (e[0] > e[1]) {  
 if (e[2] > e[0])  
 e\_high = 2;  
 else  
 e\_high = 0;  
 } else {  
 if (e[2] > e[1])  
 e\_high = 2;  
 else  
 e\_high = 1;  
 }  
  
 if (e[0] < e[1]) {  
 if (e[2] < e[0])  
 e\_low = 2;  
 else  
 e\_low = 0;  
 } else {  
 if (e[2] < e[1])  
 e\_low = 2;  
 else  
 e\_low = 1;  
 }  
  
 System.*out*.print("Наибольшее абсолютное: " + e\_1[e\_high] + "\n");  
 System.*out*.print("Наименьшее абсолютное: " + e\_1[e\_low]);

}}

## 2.3. Тестирование работы программы

Для проверки решения задача с некоторыми входными данными была прорешена в MS Excel и составлена таблица соответствия 2 решений:

Таблица 1

Тестирование работы программы и проверка результатов решения

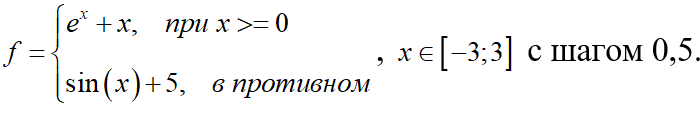
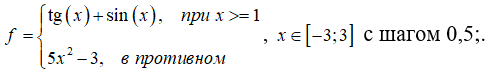
|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка MS Excel |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |

При тестировании программы на некоторых реальных и произвольных данных при правильном вводе данных ошибок выявлено не было. Сравнение проверки и ответа, который выдал код показывают правильность написанного.

# Задание 3

## 3.1. Постановка задачи

Дан произвольный **x**. Вычислить **F** если:



## 3.2. 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 x, y;

double f;   
 for (x = -3; x <= 3; x += 0.5) {  
 if (x >= 1) {  
 f = *tan*(x) + *sin*(x);  
 } else {  
 f = 5 \* *pow*(x, 2) - 3;  
 }  
 System.*out*.printf("при x = %.1f \tf = %.3f", x, f);   
 }

}}

## 3.3. Проверка решения

Таблица 1

Тестирование работы программы и проверка результатов решения

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

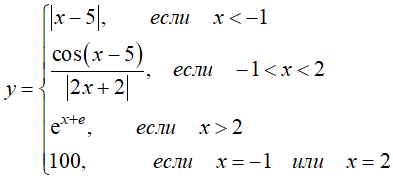
Тестирование работы программы и проверка результатов решения

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

# Задание 4

## 4.1. Постановка задачи

Дан произвольный **x**. Вычислить **Y** если:



## 4.2. 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*);   
 System.*out*.print("Введите парамеры функции: x = ");  
 x = in.nextDouble();  
  
 if (x < -1) {  
 y = *abs*(x - 5);  
 } else if (x > -1 && x < 2) {  
 y = (*cos*(x - 5)) / *abs*(2 \* x - 2);  
 } else if (x > 2) {  
 y = *exp*(x + *exp*(1));  
 } else {  
 y = 100;  
 }  
  
 System.*out*.printf("y = %.3f", y);

}}

## 4.3. Проверка решения

Таблица 1

Тестирование работы программы и проверка результатов решения

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

Тестирование работы программы и проверка результатов решения в MS Excel. В ходе небольшого количества проверок никаких ошибок выявлено не выявлено, программа работала корректно со всеми контрольными данными.

# Задание 5

## 5.1. Постановка задачи

Даны две фигуры (см. рис. 1), организовать программу, которая дает ответ, попала ли произвольно введенная точка в одну из фигур и если попала, то в какую.



## 5.2. 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*);  
 int k;  
 double x, y;

System.*out*.println("Введите координаты точки:");  
 System.*out*.print("x = ");  
 x = in.nextDouble();  
 System.*out*.print("y = ");  
 y = in.nextDouble();  
  
 // x1, y1, x2, y2  
 if ((x <= -6 && *tru\_line\_y*(1, x, y, -6, -4, -7, 5) && *tru\_line\_y*(-1, x, y, -7, 5, 1, 4))  
 || (x >= -6 && x <= -4 && *tru\_line\_y*(-1, x, y, -7, 7, 1, 4) && *tru\_line\_y*(1, x, y, -6, -4, -2, -6))  
 || (x >= -4 && *tru\_line\_y*(-1, x, y, -2, -6, -4, 0) && *tru\_line\_y*(1, x, y, -2, -6, -6, -4))  
 || (x >= -4 && *tru\_line\_y*(-1, x, y, -7, 5, 1, 4) && *tru\_line\_y*(1, x, y, -4, 0, 1, 4))) {  
 System.*out*.println("Фигура 1");  
 } else if ((x >= 1 && x <= 3 && *tru\_line\_y*(-1, x, y, 1, -2, 3, -1) && *tru\_line\_y*(1, x, y, 1, -7, 3, -4))  
 || (x >= 3 && *tru\_line\_y*(-1, x, y, 3, -1, 5, 2) && *tru\_line\_y*(-1, x, y, 5, 2, 6, -3) && *tru\_line\_y*(1, x, y, 6, -3, 3, -4))) {  
 System.*out*.println("Фигура 2");  
 } else {  
 System.*out*.println("Ха, не попал!");  
 }

}

public static boolean tru\_line\_y(double up\_doun, double x, double y, double x1, double y1, double x2, double y2) {  
 // up\_doun = 1 если область выше прямой  
 // = -1 если область ниже прямой  
 boolean out;  
 if (*abs*(up\_doun) == up\_doun) {  
 if (y >= y1 + (x - x1) \* (y2 - y1) / (x2 - x1)) {  
 out = true;  
 } else {  
 out = false;  
 }  
 } else {  
 if (y <= y1 + (x - x1) \* (y2 - y1) / (x2 - x1)) {  
 out = true;  
 } else {  
 out = false;  
 }  
 }  
 return out;  
}}

## 5.3. Проверка решения

Проверка проводилась вручную, отмечая точки на чертеже фигур.

Тестирование работы программы и проверка результатов решения

Таблица 1

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Ручная проверка |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |

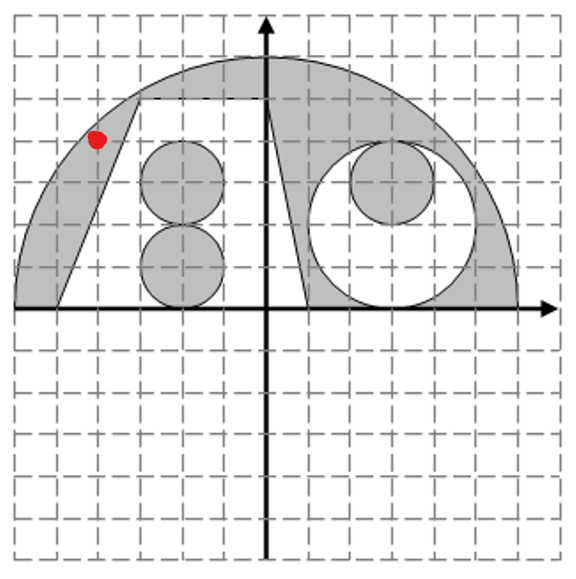
При тестировании программы на некоторых реальных и произвольных данных при правильном вводе данных ошибок выявлено не было. Сравнение ручной проверки, ответа MS Excel и ответа, который выдал код, показывают правильность написанного кода.   
Но программа некорректно работает с значениями на границах фигур, но для этого нужно дописывать отдельную проверку на попадание в границу в начале программы.

|  |  |  |
| --- | --- | --- |
| № п.п. | Ответ MS excel | Полное решение MS Excel |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |

# Задание 6

## 6.1. Постановка задачи

Пользователь вводит координаты точки с клавиатуры, определить попала ли точка в закрашенную часть фигуры, вывести сообщение на экран. Одно деление сетки соответствует 1.



## 6.2. 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*);  
 int k;  
 double x, y;

System.*out*.println("Введите координаты точки:");  
 System.*out*.print("x = ");  
 x = in.nextDouble();  
 System.*out*.print("y = ");  
 y = in.nextDouble();  
  
 if (*circle\_tru*(1, x, y, -3, 1, 1)  
 || *circle\_tru*(1, x, y, -2, 1, 1)  
 || *circle\_tru*(1, x, y, -2, 3, 1)) {  
 System.*out*.print("Молодец!");  
 } else if ((*circle\_tru*(1, x, y, 0, 0, 6) && x >= 0 && *tru\_line\_y*(1, x, y, -5, 0, -3, 5))  
 || (*circle\_tru*(1, x, y, 0, 0, 6) && x >= 5)  
 || (*circle\_tru*(1, x, y, 0, 0, 6) && x >= 0 && *circle\_tru*(-1, x, y, 3, 2, 2) && *tru\_line\_y*(1, x, y, 1, 0, 0, 4))) {  
 System.*out*.print("Ничоси, попал!");  
 } else {  
 System.*out*.print("Облом!");  
 }

}

public static boolean tru\_line\_y(double up\_doun, double x, double y, double x1, double y1, double x2, double y2) {  
 // up\_doun = 1 если область выше прямой  
 // = -1 если область ниже прямой  
 boolean out;  
 if (*abs*(up\_doun) == up\_doun) {  
 if (y >= y1 + (x - x1) \* (y2 - y1) / (x2 - x1)) {  
 out = true;  
 } else {  
 out = false;  
 }  
 } else {  
 if (y <= y1 + (x - x1) \* (y2 - y1) / (x2 - x1)) {  
 out = true;  
 } else {  
 out = false;  
 }  
 }  
 return out;  
}

public static boolean circle\_tru(int in\_out, double x, double y, double x1, double y1, double r) {  
 boolean out;  
 if (*abs*(in\_out) == in\_out) {  
 if (r >= *sqrt*(*pow*(x - x1, 2) + *pow*(y - y1, 2))) {  
 out = true;  
 } else {  
 out = false;  
 }  
 } else {  
 if (r < *sqrt*(*pow*(x - x1, 2) + *pow*(y - y1, 2))) {  
 out = true;  
 } else {  
 out = false;  
 }  
 }  
 return out;  
}}

## 6.3. Проверка решения

Проверка проводилась вручную, отмечая точки на чертеже фигур.

Тестирование работы программы и проверка результатов решения

Таблица 1

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Ручная проверка |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |

При тестировании программы на некоторых реальных и произвольных данных при правильном вводе данных ошибок выявлено не было. Сравнение ручной проверки и ответа, который выдал код показывают правильность написанного кода.

|  |  |  |
| --- | --- | --- |
| № п.п. | Ответ MS Excel | Полное решение MS Excel |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |

# Задание 7

## 7.1. Постановка задачи

Составить две программы для определения функций с точками разрыва. Программа должна учитывать, что пользователь может ввести переменную, которая не удовлетворяет интервалу определения функции . Если, введенный аргумент из области определения функции не подходит из-за особенностей аргументов математических функций для корректного определения функции, то программа должна сообщить об этом.

## 7.2. 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 g, z;  
 System.*out*.print("Введите x = ");  
 x = in.nextDouble();  
 if (x < -4 || x > 4) {  
 System.*out*.print("Вы ввели число, не входящая в диапазон x = [-4,4]");  
 break;  
}  
 if (x % *PI* / 2 != 0 && x % *PI* \* 3 / 2 != 0 && 10 + *pow*(*tan*(x), 3) != 0 && 1 + 2 \* *sin*(x) >= 0) {  
 if (x <= 0) {  
 g = 2 \* *pow*(x, 2) + (3 \* x + *pow*(*sin*(2 \* x), 2 \* *E*)) / (10 + *pow*(*tan*(x), 3));  
 } else {  
 g = 2 \* *E* + *sqrt*(1 + 2 \* *sin*(x));  
 }  
 } else {  
 System.*out*.print("Введённый x не входит в область определения функции 'g'");  
 break;  
 }  
  
 if (*E* + *pow*(x, 2) != 0 && *cos*(2 \* x + 1) + *exp*(*pow*(x, 2)) / (*E* + *pow*(x, 2))>=0  
 && x - 2 != 0 && x != 0 && 1 / (x - 2) + *abs*(2 \* *sin*(*pow*(3 \* x, 1 / 4)))>=0){  
 if (x < 0) {  
 z = *sqrt*(*cos*(2 \* x + 1) + *exp*(*pow*(x, 2)) / (*E* + *pow*(x, 2)));  
 } else if (x >= 0 && x <= 1) {  
 z = 2 \* *pow*(*cos*(*pow*(*sin*(x), 3)), 2);  
 } else {  
 z = *sqrt*(1 / (x - 2) + *abs*(2 \* *sin*(*pow*(3 \* x, 1.0 / 4.0))));  
 }  
 } else {  
 System.*out*.print("Введённый x не входит в область определения функции 'z'");  
 break;  
 }  
  
 if (g != g) {  
 System.*out*.print("g - Система уравнеий не имеет решений при данном x \n");  
 } else {  
 System.*out*.printf("g = %.4f \n", g);  
 }  
 if (z != z) {  
 System.*out*.print("Система уравнеий z не имеет решений при данном x");  
 } else {  
 System.*out*.printf("z = %.4f", z);  
 }

}}

## 7.3. Проверка решения

Таблица 1

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 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.

Во ячейке формулы g:

=ЕСЛИ(И($C$2>=-4;$C$2<=4);  
ЕСЛИОШИБКА(  
ЕСЛИ($C$2<=0;  
2\*(C2^2)+3\*$C$2+СТЕПЕНЬ(SIN(2\*$C$2);2\*EXP(1))/(10+СТЕПЕНЬ(TAN($C$2);3));  
2\*EXP(1)+КОРЕНЬ(1+2\*SIN($C$2)));  
"система не имеет решения при данном X");  
"данный не входит в интервал определения функции [-4;4]")

В ячейке z:

=ЕСЛИ(И($C$2>=-4;$C$2<=4);  
ЕСЛИОШИБКА(  
ЕСЛИ(И($C$2>=0;C2<=1);   
2\*(SIN( COS(C2)^2 )^2 );   
ЕСЛИ(C2<0;   
КОРЕНЬ(COS(2\*C2+1) + EXP(C2^2)/(EXP(1) + C2^2));   
КОРЕНЬ(1/(C2-2) + ABS(2 \* SIN(СТЕПЕНЬ(3\*C2; 1/4)))) ) );  
"система не имеет решения при данном X");  
"данный не входит в интервал определения функции [-4;4]")

# Задание 8

## 8.1. Постановка задачи

Используя инструкцию switch составить программу, которая выводит пользователю скорость вращения планеты вокруг солнца (для нашей солнечной системы). Пользователь вводит номер планеты, а ему выдается развернутый ответ, что это за планета и какова ее скорость вращения вокруг Солнца. Планеты пронумерованы от Солнца.

## 8.2. 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[] speed = {47.8, 35.0, 29.8, 24.1, 13.0, 9.6, 6.8, 5.4, 4.7};  
 String[] planets = {"Меркурий", "Венера", "Земля", "Марс", "Юпитер", "Уран", "Нуптун", "Плутон"};  
 System.*out*.print("Введите номер планеты: ");  
 int i = in.nextInt();  
  
 System.*out*.printf("'Планета %s %d-ая по счёту от Солнца, вращается со скоростью %.1f км/с.'", planets[i - 1], i + 1, speed[i - 1]);

}}

## 8.3. Тестирование работы программы

Таблица 1

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  | **верно** |
| 2 |  | **верно** |
| 3 |  | **верно** |

При тестировании программы на некоторых реальных и произвольных данных при правильном вводе данных ошибок выявлено не было.

Я сделал это задание через f-строки, так как этот вариант раза в 4 короче, чем switch, да и красивее выглядит. А switch я использовал в общей структуре перехода к каждому заданию.