Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчёт по лабораторной работе № 5

тема «Циклы в Java»

по дисциплине «Информатика»

Выполнил: студент группы ИСТ-22-1б Синьковский Г.С.

Пермь, 2022

**Содержание**

[Задание 1 3](#_Toc119395672)

[1.1. Постановка задачи 3](#_Toc119395673)

[1.2. Решение задачи, код программы 3](#_Toc119395674)

[1.3. Тестирование работы программы с проверкой 3](#_Toc119395675)

[Задание 2 4](#_Toc119395676)

[2.1. Постановка задачи 4](#_Toc119395677)

[2.2. Решение задачи, код программы 4](#_Toc119395678)

[2.3. Тестирование работы программы с проверкой 4](#_Toc119395679)

[Задание 3 5](#_Toc119395680)

[3.1. Постановка задачи 5](#_Toc119395681)

[3.2. Решение задачи, код программы 5](#_Toc119395682)

[3.3. Тестирование работы программы с проверкой 5](#_Toc119395683)

[Задание 4 6](#_Toc119395684)

[4.1. Постановка задачи 6](#_Toc119395685)

[4.2. Решение задачи, код программы 6](#_Toc119395686)

[4.3. Тестирование работы программы с проверкой 6](#_Toc119395687)

[Задание 5 7](#_Toc119395688)

[5.1. Постановка задачи 7](#_Toc119395689)

[5.2. Решение задачи, код программы 7](#_Toc119395690)

[5.3. Тестирование работы программы с проверкой 7](#_Toc119395691)

[Задание 6 8](#_Toc119395692)

[6.1. Постановка задачи 8](#_Toc119395693)

[6.2. Решение задачи, код программы 8](#_Toc119395694)

[6.3. Тестирование работы программы с проверкой 8](#_Toc119395695)

[Задание 7 9](#_Toc119395696)

[7.1. Постановка задачи 9](#_Toc119395697)

[7.2. Решение задачи, код программы 9](#_Toc119395698)

[7.3. Тестирование работы программы с проверкой 9](#_Toc119395699)

[Задание 8 10](#_Toc119395700)

[8.1. Постановка задачи 10](#_Toc119395701)

[8.2. Решение задачи, код программы 10](#_Toc119395702)

[8.3. Тестирование работы программы с проверкой 10](#_Toc119395703)

[Задание 9 11](#_Toc119395704)

[9.1. Постановка задачи 11](#_Toc119395705)

[9.2. Решение задачи, код программы 11](#_Toc119395706)

[9.3. Тестирование работы программы с проверкой 11](#_Toc119395707)

[Задание 10 12](#_Toc119395708)

[10.1. Постановка задачи 12](#_Toc119395709)

[10.2. Решение задачи, код программы 12](#_Toc119395710)

[10.3. Тестирование работы программы с проверкой 12](#_Toc119395711)

# Задание 1

## 1.1. Постановка задачи

1. Найти сумму первых N членов ряда и найти сумму членов ряда, которые меньше заданного с клавиатуры числа M:



## 1.2. Решение задачи, код программы

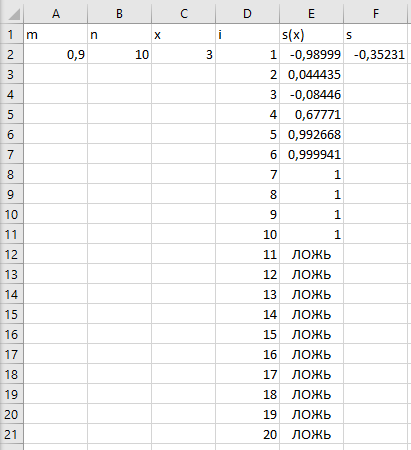
import java.util.\*;  
import static java.lang.Math.\*;  
public class Zadanie\_1 {  
 public static long factorio(int x) {  
 long r=1;  
 while (x>0) {  
 r\*=x;  
 x--;  
 }  
 return r;  
 }  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 int i=1;  
 double s=0;  
 System.*out*.println("Введите M");  
 double m = in.nextDouble();  
 System.*out*.println("Введите N");  
 double n = in.nextDouble();  
 System.*out*.println("Введите x");  
 double x = in.nextDouble();  
 for (i=1; i<n; i++) {  
 if(*pow*(*cos*(*pow*(x,2\*i-1)/*factorio*(2\*i-1)),i)<m) {  
 s+=*pow*(*cos*(*pow*(x,2\*i-1)/*factorio*(2\*i-1)),i);  
 }  
 }  
 System.*out*.println(s);  
 }  
}

## 1.3. Тестирование работы программы с проверкой

Формулы для вычисления:

E2:E21 =ЕСЛИ(D2<=$B$2;COS(($C$2^(2\*D2-1))/(ФАКТР(2\*D2-1)))^(D2))

H2 = СУММЕСЛИ(E2:E21; "<"&A2)



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. Неопределенностей при решении задачи выявлено не было, возможно данная ситуация связанна с малым количеством проверок данных.

# Задание 2

## 2.1. Постановка задачи

2. Из натурального числа N удалить заданную цифру K. Число и цифру вводить с клавиатуры (Например, задано число 5683. Требуется удалить из него цифру 8. Получится число 563)

## 2.2. Решение задачи, код программы

import java.util.\*;  
public class Zadanie\_2 {  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Введите число");  
 String a = in.nextLine();  
 System.*out*.println("Введите цифру, которую нужно удалить");  
 String b = in.nextLine();  
 System.*out*.println(a.replace(b,""));  
 }  
}

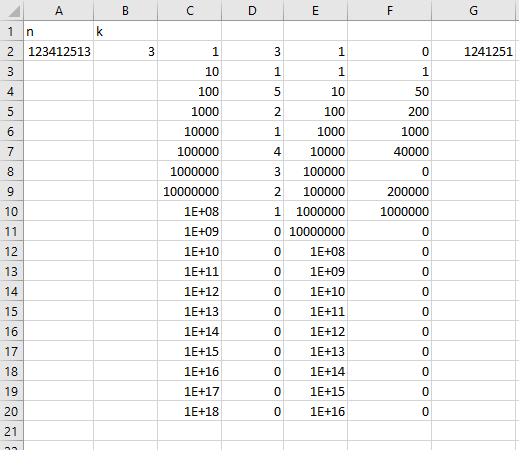
## 2.3. Тестирование работы программы с проверкой

Формулы для вычисления:

B2 = ОСТАТ(A2,10)

C2 = НАИБОЛЬШИЙ(B2:B7,1)

D2 = СЧЁТЕСЛИ(B2:B17,C2)



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

3. Написать программы, которые вычисляют выражения:

, , 

## 3.2. Решение задачи, код программы

import static java.lang.Math.\*;  
public class Zadanie\_3 {  
 public static void main(String[] args) {  
 double a=0,b=0,c=1;  
 for (int i=1; i<=8; i++) {  
 for (int j=1; j<=i; j++) {  
 a+=*pow*(j+i,2)/*cos*(j+*pow*(i,2));  
 }  
 b+=a;  
 }  
 System.*out*.println(b);  
 a=b=1;  
 for (int i=1; i<=5; i++) {  
 for (int j=1; j<=i; j++) {  
 a\*=*asin*(*pow*(j,-i));  
 }  
 b\*=a;  
 }  
 System.*out*.println(b);  
 a=b=0;  
 for (int i=1; i<=8; i++) {  
 for (int j=i; j<=2\*i-1; j++) {  
 for (int k=i+j; k<=2\*(i+j); k++) {  
 a+=k+4\*(j+*pow*(i,k));  
 }  
 b+=a;  
 }  
 c\*=b;  
 }  
 System.*out*.println(c);  
 }  
}

# Задание 4

## 4.1. Постановка задачи

4. Пользователь вводит целое десятичное число. Написать программу перевода целого десятичного числа из десятичной системы счисления в систему счисления с основанием, которое пользователь вводит с клавиатуры (вводим ограничения на системы счисления, в которых есть буквенное обозначение цифр, так же не может быть введено 0 и 1 как основание системы счисления). После перевода сделать проверку. Пользователь может выбирать систему счисления до бесконечности, т.е. необходимо предусмотреть внешний цикл с вопросом к пользователю о необходимости продолжать перевод из одной системы счисления в другую.

## 4.2. Решение задачи, код программы

import java.util.\*;  
  
public class Zadanie\_4 {  
 public static double perevod(double x,int s) {  
 int xc = (int)x;  
 double xd = x-xc;  
 String x1 = "", x2 = "0.";  
 while (xc>0) {  
 x1+=Integer.*toString*(xc%s);  
 xc/=s;  
 }  
 for (int a=0; a<10; a++) {  
 xd=xd\*s-(int)xd\*s;  
 x2+=Integer.*toString*((int)xd);  
 }  
 return Integer.*parseInt*(new StringBuilder(x1).reverse().toString())+Double.*parseDouble*(new StringBuilder(x2).toString());  
 };  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Введите дробное число");  
 double a = in.nextDouble();  
 int s=0;  
 while (s<1 | s>10) {  
 System.*out*.println("Введите систему счисления (1<=s<=10)");  
 s=in.nextInt();  
 }  
 System.*out*.println(*perevod*(a,s));  
 }  
}

## 4.3. Тестирование работы программы с проверкой

Формула для вычисления:

С2=ОСНОВАНИЕ(A2;B2;1)

D2=A2-ЦЕЛОЕ(A2)

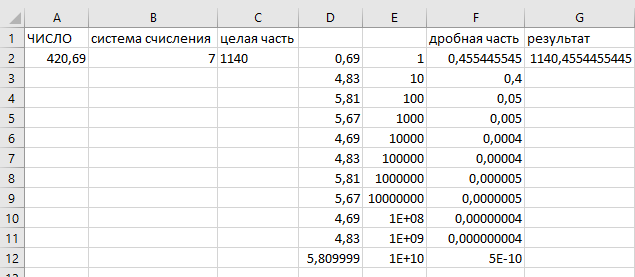
D3:D12=D2\*$B$2-ЦЕЛОЕ(D2)\*$B$2

E3:E12=E2\*10

F3:F12=ЦЕЛОЕ(D3)/E3

F2=СУММ(F3:F12)

G2=C2+F2



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 6

## 6.1. Постановка задачи

6. Переделать программу (класс) задания 6 лабораторной работы 5 о попадании точки в область в класс без метода main с названием Oblast. Метод main заменить на метод Oblast c входными данными в виде координат произвольной точки пространства и выходными данными типа boolean (true – если точка попала в область, false – если точка не попала в область). Создать программу, взаимодействующую с классом Oblast (без использования наследования), в которой пользователь в цикле проверяет попадание точек в область до бесконечности, т.е. необходимо предусмотреть цикл с вопросом к пользователю о необходимости проверки точки.

## 6.2. Решение задачи, код программы

import java.util.\*;  
import static java.lang.Math.\*;  
public class Zadanie\_6 {  
 public static boolean oblast(double x, double y) {  
 if (*pow*(x,2)+*pow*(y,2)<=36 & x>=0 & !(y<=4 & y>=x+1) &  
 !(y<=-4f/3\*x+1 & y>=1f/3\*x-4) & !(*pow*(x-4,2) + *pow*(y+2,2)<=1) &  
 !((*pow*(x-3,2) + *pow*(y-1,2)<=4) & !(*pow*(x-2,2)+*pow*(y-1,2)<=1))) {  
 return true;  
 }  
 else {  
 return false;  
 }  
 }  
 public static void main(String[] args) {  
 double x=0, y=0;  
 Scanner in = new Scanner(System.*in*);  
 boolean stop=false;  
 while (!stop) {  
 System.*out*.println("Введите \"stop\", если хотите прекратить поиск координат.");  
 String vvod = in.nextLine();  
 if (vvod.equals("stop")) {  
 stop = true;  
 } else {  
 System.*out*.println("Введите координаты х");  
 x = Double.*parseDouble*(in.nextLine());  
 System.*out*.println("Введите координаты y");  
 y = Double.*parseDouble*(in.nextLine());  
 if (*oblast*(x, y)) {  
 System.*out*.println("Точка попала в область");  
 } else {  
 System.*out*.println("Точка не попала в область");  
 }  
 }  
 }  
 }  
}

## 6.3. Тестирование кода программы

## 

# Задание 7

## 7.1. Постановка задачи

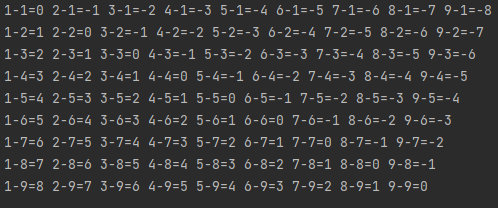
7. С использованием циклов сформировать вывод полной таблицы вычитания в виде:

|  |  |  |  |
| --- | --- | --- | --- |
| 1 – 1 = 0 | 2 – 1 = 1 | ... | 9 – 1 = 8 |
| 1 – 2 = –1 | 2 – 2 = 0 | ... | 9 – 2 = 7 |
| ... | ... | ... | ... |
| 1 – 9 = –8 | 2 – 9 = –7 | ... | 9 – 9 = 0 |

## 7.2. Решение задачи, код программы

public class Zadanie\_7 {  
 public static void main(String[] args) {  
 for (int i = 1; i < 10; i++) {  
 for (int j = 1; j < 10; j++) {  
 System.*out*.print(j+"-"+i+"="+(i-j)+" ");  
 }  
 System.*out*.println("");  
 }  
 }  
}

## 7.3. Тестирование кода программы



# Задание 8

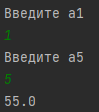
## 8.1. Постановка задачи

8. Переделать программу (класс) задания 1 лабораторной работы 4 о нахождении значений 2-х функций в класс без метода main с названием FunctionMy. Метод main заменить на метод FunctionMy c входными данными. Создать программу, взаимодействующую с классом FunctionMy, в которой пользователь в цикле находит сумму 10 значений функции изменяя только один параметр функции в цикле, остальные параметры, которые входя в формулу, считаются константами. Взаимодействие с классом FunctionMy сделать в виде наследования.

## 8.2. Решение задачи, код программы

import java.util.Scanner;  
public class Zadanie\_8 {  
 public static double FunctionMy(double a1, double razn) {  
 int s=0;  
 for (int i=1; i<=10; i++) {  
 s+=a1;  
 a1+=razn;  
 }  
 return s;  
 }  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
 System.*out*.println("Введите a1");  
 double a1 = (sc.nextInt());  
 System.*out*.println("Введите a5");  
 double a5 = sc.nextInt();  
 double razn=(a5-a1)/4;  
 System.*out*.println(*FunctionMy*(a1,razn));  
 }  
}

## 8.2. Тестирование кода программы



# Задание 9

## 9.1. Постановка задачи

9. Написать программу вычисление корня р-й степени (степень вводиться с клавиатуры) в рамках итерационной процедуры . Для определения используется итерационная процедура на основе формулы Ньютона , , при этом . Остановка итерационной процедуры , где  – точность вычисления. В рамках программы определить число итраций, которые потребовались для отыскания корня р-й степени в рамках цикла с параметром для точности от 10-2 до 10-6, шаг 10-1. Организовать форматированный вывод результатов в виде: Точность Корень Число итераций.

## 9.2. Решение задачи, код программы

import java.util.\*;  
import static java.lang.Math.\*;  
  
public class Zadanie\_9 {  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Введите x");  
 double x = (in.nextDouble());  
 System.*out*.println("Введите p");  
 double p = (in.nextDouble());  
 double y = *log*(x\*(p+1))/p;  
 int c=0;  
 for (double e=*pow*(10,-1); e>*pow*(10,-6); e\*=0.1) {  
 while (true) {  
 double ly=y;  
 y=1/p\*((p-1)\*y+x/*pow*(y,(p-1)));  
 c++;  
 if (*abs*(y-ly)<e) {  
 System.*out*.println("Точность: "+e+"\nКорень: "+y  
 +"\nЧисло итераций: "+c+"\n");  
 break;  
 }  
 }  
 }  
 }  
}

# Задание 10

## 10.1. Постановка задачи

10. Разработать алгоритм приближённого вычисления площади криволинейной фигуры, ограниченной осью абсцисс, графиком заданной функции , и вертикальными прямыми, т.е. . Каждый отрезок функции представляется в виде прямоугольника c длиной отрезка (шагом)  (см. рис. ниже). Затем площадь под кривой вычисляется по формуле  – центральный прямоугольник, т.к. высота прямоугольника берется как значение функции в центре каждого прямоугольника. Вычислить значения площади под кривой при n равном 10, 100, 1000, 10000 в рамках цикла по n. Оценить погрешность решения при разных шагах по сравнению с точным аналитическим решением. Организовать форматированный вывод результатов в виде:

n h S Аналитическое решение Погрешность.

|  |  |
| --- | --- |
|  |  |
| а | б |

Рис. Разбиение площади под кривой на прямоугольники с шагом h:

а – общий вид; б – i-й центральный прямоугольник

## 10.2. Решение задачи, код программы

import static java.lang.Math.\*;  
  
public class Zadanie\_10 {  
 public static void main(String[] args) {  
 double sum;  
 double h;  
 for(double n = 10;n <= 10000; n \*= 10){  
 h = 1/n;  
 sum = 0;  
 for (int i = 0; i < n; i++){  
 sum += (20\**pow*(n-h\*i,2)-5\**pow*(*sin*(n-h\*i)+(n-h\*i),2)) \* h;  
 }  
 System.*out*.println(n + "\n" + h + "\n" + sum + "\n5.18\n" + *abs*(sum - 5.18)+ "\n");  
 }  
  
 }

