МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Санкт-Петербургский национальный исследовательский университет

информационных технологий, механики и оптики»

ФАКУЛЬТЕТ СИСТЕМ УПРАВЛЕНИЯ И РОБОТОТЕХНИКИ

**ЛАБОРАТОРНАЯ РАБОТА No1**

по дисциплине

«ПРОГРАММИРОВАНИЕ»

Вариант No 368823

**Выполнил**:

Студент группы P3112

Соколов Анатолий

Владимирович

**Преподаватель**:

Письмак Алексей

Евгеньевич

Санкт-Петербург, 2022

Содержание

Задание ......................................................................................................... 3

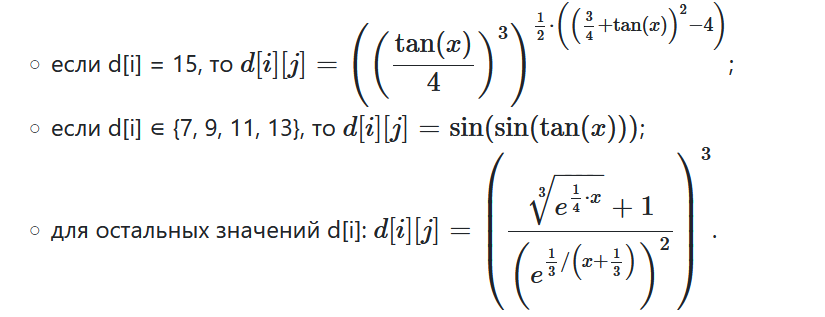
Исходный код программы........................................................................... 4

Результаты работы программы................................................................... 6

Вывод............................................................................................................. 7

Задание

1. Создать одномерный массив d типа long. Заполнить его нечётными числами от 5 до 19 включительно в порядке убывания.
2. Создать одномерный массив x типа float. Заполнить его 16-ю случайными числами в диапазоне от -5.0 до 14.0.
3. Создать двумерный массив d размером 8x16. Вычислить его элементы по следующей формуле (где x = x[j]):



1. Напечатать полученный в результате массив в формате с двумя знаками после запятой.

Исходный код программы

public class Main {  
// first task  
 public static int sum\_var = 19;  
 public static final long[] single\_dim\_array\_d = new long[10];  
 public static void first\_task() {  
 for (int i = 0; i < 10; i++) {  
 single\_dim\_array\_d[i] += sum\_var;  
 sum\_var -= 2;  
 System.out.print(single\_dim\_array\_d[i] + " ");  
 }  
 System.out.println();  
 }  
// second task  
 public static final float[] single\_dim\_array\_x = new float[16];  
 public static void second\_task() {  
 for (int i = 0; i < single\_dim\_array\_x.length; i++) {  
 single\_dim\_array\_x[i] = (float) (-5 + (Math.random() \* (29)));  
 }  
 for (int i = 0; i < single\_dim\_array\_x.length; i++) {  
 if (single\_dim\_array\_x[i] > 14) {  
 single\_dim\_array\_x[i] -= 10;  
 };  
 System.out.printf("%.2f ", (single\_dim\_array\_x[i]));  
 }  
 System.out.println();  
 }  
// third task  
public static final double[][] two\_dim\_array = new double[8][16];  
public static void third\_task() {  
 for (int i = 0; i < 8; i++) {  
 for (int j = 0; j < 16; j++) {  
 if (single\_dim\_array\_d[i] == 15) {  
 two\_dim\_array[i][j] = Math.pow((Math.pow((Math.tan(single\_dim\_array\_x[j]) / 4),3)),  
 (0.5 \* ((Math.pow((0.75 + Math.tan(single\_dim\_array\_x[j])), 2)) - 4)));  
 } else if (single\_dim\_array\_d[i] == 7 || single\_dim\_array\_d[i] == 9  
 || single\_dim\_array\_d[i] == 11 || single\_dim\_array\_d[i] == 13) {  
 two\_dim\_array[i][j] = Math.sin(Math.sin(Math.tan(single\_dim\_array\_x[j])));  
 } else {  
 two\_dim\_array[i][j] = ((1 + Math.cbrt(Math.pow(Math.E, (single\_dim\_array\_x[j] / 4))))  
 / (Math.pow(Math.E, (2 / (1 + 3 \* single\_dim\_array\_x[j])))));  
 }  
 if (two\_dim\_array[i][j] != (1.0 / 0.0) & two\_dim\_array[i][j] != (-1.0 / 0.0) & (  
 two\_dim\_array[i][j] > 0 || two\_dim\_array[i][j] <= 0)) {  
 System.out.printf("%.2f ", two\_dim\_array[i][j]);  
 }  
 }  
 }  
}

public static void main(String[] args){  
 first\_task();  
 second\_task();  
 third\_task();  
 }  
}

19 17 15 13 11 9 7 5 3 1

11.13 -0.92 13.94 2.45 7.32 -4.09 12.19 4.69 -4.17 -0.93 4.45 4.25 13.22 5.74 1.68 7.09

3.33 5.99 4.00 1.75 2.60 2.04 3.57 2.17 2.03 5.89 2.13 2.10 3.82 2.34 1.54 2.56 3.33 5.99 4.00 1.75 2.60 2.04 3.57 2.17 2.03 5.89 2.13 2.10 3.82 2.34 1.54 2.56 12071.92 0.08 0.12 0.03 66.96 5.24 -0.81 -0.82 -0.82 -0.67 0.84 -0.83 -0.37 -0.02 -0.84 -0.83 -0.47 0.79 0.64 -0.54 -0.21 0.76 -0.81 -0.82 -0.82 -0.67 0.84 -0.83 -0.37 -0.02 -0.84 -0.83 -0.47 0.79 0.64 -0.54 -0.21 0.76 -0.81 -0.82 -0.82 -0.67 0.84 -0.83 -0.37 -0.02 -0.84 -0.83 -0.47 0.79 0.64 -0.54 -0.21 0.76 -0.81 -0.82 -0.82 -0.67 0.84 -0.83 -0.37 -0.02 -0.84 -0.83 -0.47 0.79 0.64 -0.54 -0.21 0.76 3.33 5.99 4.00 1.75 2.60 2.04 3.57 2.17 2.03 5.89 2.13 2.10 3.82 2.34 1.54 2.56

Process finished with exit code 0

Вывод

Язык программирования (далее ЯП) java способен работать с простыми числами, числами с плавающей запятой и разными библиотеками. ЯП также способен создавать массивы из этих чисел или символов. Для создания массивов необходимо выделять определенно количество памяти. ЯП способен выводить бесконечность в качестве результата действий и работать с ней, а также автоматически определять: является ли ответ вещественным числом или нет. ЯП является относительно низкоуровневым, благодаря чему достигается высокая скорость исполнения команд.