

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники (ФПИиКТ)

Программирование

Лабораторная работа №1

Вариант 311605

Выполнил

Григорьев Даниил Александрович

Группа 3116

Санкт-Петербург 2024

## Текст задания

Введите вариант:

1. Создать одномерный массив  $z$  типа `short`. Заполнить его нечётными числами от 5 до 23 включительно в порядке убывания.
2. Создать одномерный массив  $x$  типа `float`. Заполнить его 12-ю случайными числами в диапазоне от -2.0 до 15.0.
3. Создать двумерный массив  $z$  размером  $10 \times 12$ . Вычислить его элементы по следующей формуле (где  $x = x[j]$ ):

◦ если  $z[i] = 13$ , то  $z[i][j] = \frac{\sqrt[3]{\sin(x)} + 1}{2}$ ;

◦ если  $z[i] \in \{11, 15, 17, 21, 23\}$ , то  $z[i][j] = \arcsin \left( \frac{1}{\sin^2 \left( \arcsin \left( \frac{x+6.5}{17} \right) \right) \frac{\arcsin \left( \frac{x+6.5}{17} \right)}{2}} \right)$ ;

◦ для остальных значений  $z[i]$ :  $z[i][j] = e^{\cos(e^x)}$ .

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

## Исходный код программы

```
import static java.lang.Math.*;

public class Main {
    public static final short[] z = new short[10];
    public static final float[] x = new float[12];
    public static double e = 2.71828182845904;

    public static void main(String[] args){
        int c = 0;
        for (int i = 23; i >= 5; i-=2){
            z[c] = (short)i;
            c += 1;
        }

        for (int i = 0; i < 12; ++i){
            x[i] = getRandomValue();
        }
        workWithArray(z, x);
    }

    public static float getRandomValue(){
        float value = (float)random()*17-2;
        return value;
    }

    public static void workWithArray(short[] firstz, float[] firstx){
        float x;
        float[][] z1;
        z1 = new float[10][12];
        for (int i = 0; i < 10; i++){
            for (int j = 0; j < 12; j++){
                x = firstx[j];
                switch(firstz[i]){
                    case(13): z1[i][j] = (float)((pow(sin(x), 1.0/3) + 1)/2);
                    break;
                    case(11):
                    case(15):
                    case(17):
                    case(21):
                    case(23):
                        z1[i][j] = (float)(asin(1/pow(e, (pow(sin(pow((asin((x + 6.5)/17)), (asin((x + 6.5)/17))/2)),
2)))))); break;
                    default: z1[i][j] = (float)pow(e, cos(pow(e, pow(e, x)))); break;
                }
            }
        }
        printArray(z1);
    }

    public static void printArray(float[][] array){
        for (int i = 0; i < 10; i++){
            System.out.println(" ");
            for (int j = 0; j < 12; ++j){
                double answer = array[i][j];
                if (Double.isNaN(answer)){
                    System.out.print(" |");
                    System.out.print("NaN");
                    System.out.print(" | ");
                }
                else{
                    System.out.print(" |");
                    System.out.format("%.2f", answer);

                    System.out.print(" | ");
                }
            }
        }
    }
}
```

# Результат работы программы

1 запуск:

0,62	0,62	0,62	0,59	0,62	0,48	NaN	0,62	0,61	0,51	0,62	0,51
0,62	0,62	0,62	0,59	0,62	0,48	NaN	0,62	0,61	0,51	0,62	0,51
0,89	0,96	0,37	1,36	1,25	NaN	NaN	2,68	1,50	NaN	0,37	NaN
0,62	0,62	0,62	0,59	0,62	0,48	NaN	0,62	0,61	0,51	0,62	0,51
0,62	0,62	0,62	0,59	0,62	0,48	NaN	0,62	0,61	0,51	0,62	0,51
NaN	NaN	0,95	NaN	NaN	0,87	0,98	0,92	NaN	1,00	0,95	1,00
0,62	0,62	0,62	0,59	0,62	0,48	NaN	0,62	0,61	0,51	0,62	0,51
0,89	0,96	0,37	1,36	1,25	NaN	NaN	2,68	1,50	NaN	0,37	NaN
0,89	0,96	0,37	1,36	1,25	NaN	NaN	2,68	1,50	NaN	0,37	NaN
0,89	0,96	0,37	1,36	1,25	NaN	NaN	2,68	1,50	NaN	0,37	NaN

2 запуск:

0,60	0,62	0,60	NaN	0,62	0,46	0,47	0,58	0,58	0,59	0,62	0,54
0,60	0,62	0,60	NaN	0,62	0,46	0,47	0,58	0,58	0,59	0,62	0,54
0,38	2,39	0,60	NaN	0,66	NaN	NaN	2,68	1,85	0,62	0,44	NaN
0,60	0,62	0,60	NaN	0,62	0,46	0,47	0,58	0,58	0,59	0,62	0,54
0,60	0,62	0,60	NaN	0,62	0,46	0,47	0,58	0,58	0,59	0,62	0,54
0,78	0,91	NaN	0,97	NaN	NaN	0,81	NaN	NaN	NaN	0,94	0,93
0,60	0,62	0,60	NaN	0,62	0,46	0,47	0,58	0,58	0,59	0,62	0,54
0,38	2,39	0,60	NaN	0,66	NaN	NaN	2,68	1,85	0,62	0,44	NaN
0,38	2,39	0,60	NaN	0,66	NaN	NaN	2,68	1,85	0,62	0,44	NaN
0,38	2,39	0,60	NaN	0,66	NaN	NaN	2,68	1,85	0,62	0,44	NaN

3 запуск:

0,55	0,58	0,44	0,51	0,55	NaN	0,62	0,61	0,62	NaN	NaN	0,50
0,55	0,58	0,44	0,51	0,55	NaN	0,62	0,61	0,62	NaN	NaN	0,50
0,40	1,05	NaN	NaN	2,39	NaN	0,95	1,46	0,41	NaN	NaN	NaN
0,55	0,58	0,44	0,51	0,55	NaN	0,62	0,61	0,62	NaN	NaN	0,50
0,55	0,58	0,44	0,51	0,55	NaN	0,62	0,61	0,62	NaN	NaN	0,50
NaN	NaN	NaN	1,00	0,60	NaN	NaN	NaN	0,81	NaN	NaN	0,98
0,55	0,58	0,44	0,51	0,55	NaN	0,62	0,61	0,62	NaN	NaN	0,50
0,40	1,05	NaN	NaN	2,39	NaN	0,95	1,46	0,41	NaN	NaN	NaN
0,40	1,05	NaN	NaN	2,39	NaN	0,95	1,46	0,41	NaN	NaN	NaN
0,40	1,05	NaN	NaN	2,39	NaN	0,95	1,46	0,41	NaN	NaN	NaN

## Вывод

Я написал программу, руководствуясь техническим заданием, выполняющую указанные в варианте действия. Программа использует математические функции из стандартной библиотеки Java. Вычисление очередного элемента двумерного массива реализовано в виде отдельного статического метода. Результат вычисления выражения выводится в стандартный поток вывода в виде матрицы с элементами в указанном в варианте формате. Вывод матрицы реализован в виде отдельного статического метода. Программа упакована в исполняемый jar-архив.