

**Министерство науки и высшего образования Российской  
Федерации**

**Федеральное государственное автономное образовательное  
учреждение высшего образования**

**«Национальный исследовательский университет ИТМО»**

**Факультет информационных технологий и программирования**

**Лабораторная работа № 10 Вариант 11**

*Название работы: Функции*

**Выполнил студент группы № М3113**

**Крамской Вадим Вадимович**

**Подпись:**



**Санкт-Петербург  
2022**

Условие:

1. Реализовать функции нахождения наибольшего общего делителя и наименьшего общего кратного двух натуральных чисел.
2. На плоскости заданы своими координатами  $n$  точек. Разработать функцию, создающую массив размером  $n \times (n - 1)$ , элементами которого являются расстояния от каждой из точек до  $n - 1$  других.

Решение:

1) Для реализации задачи был выбран один из нескольких алгоритмов нахождения НОД и НОК, а также добавлены вспомогательный функции `min` и `max`.

```
#include <stdio.h>

int max(int chis1, int chis2){
    if (chis1-chis2 < 0) {
        return(chis2);
    } else {
        return(chis1);
    }
}

int min(int chis1, int chis2){
    if (chis1-chis2 < 0) {
        return(chis1);
    } else {
        return(chis2);
    }
}

int NOD (int chis1, int chis2){
    int maxc = max(chis1,chis2);
    int minc = min(chis1,chis2);
    int del1 = maxc % minc;
    if (del1 ==0) {
        return minc;
    }
    int del2 = minc % del1;
    if (del2 ==0){
        return del1;
    }
    while (min( chis1: del1, chis2: del2) != 0 || del1 % del2 != 0) {
        if (del2 != 0) {
            del1 = del1 % del2;
        }
        if (del1 != 0) {
            del2 = del2 % del1;
        }
    }
    if (del2 ==0) {
        return (del1);
    } else {
        return (del2);
    }
}
```

```

        return (del2);
    }
}

int NOK (int chis1, int chis2){
    int del = NOD(chis1,chis2);
    int proiz = chis1*chis2;
    return (proiz/del);
}

int main() {
    int chis1;
    int chis2;
    scanf( format: "%d",&chis1);
    scanf( format: "%d",&chis2);
    printf( format: "%d, %d",NOD(chis1,chis2),NOK(chis1,chis2));
    return 0;
}

```

2) На вход было получено n координат точек, после чего в функции zap с помощью цикла был заполнен двумерный массив, который содержит расстояние от каждой до каждой точки.

```

#include <stdio.h>
#include <math.h>

void zap(int n, int rast[n][n-1], int koordx[], int koordy[]){
    int count = 0;
    for(int i = 0; i < n; ++i){
        for(int j = 0; j < n; ++j){
            if (i!=j) {
                rast[count][i] = sqrt( (X: pow( X: koordx[i] - koordx[j], Y: 2) + pow( X: koordy[i] - koordy[j], Y: 2));
                ++count;
            }
        }
    }
    count = 0;
}

int main() {
    int n;
    scanf( format: "%d",&n);
    int rast[n][n-1];
    int koordx[n];
    int koordy[n];
    for (int i = 0; i < n; ++i){
        scanf( format: "%d %d",&koordx[i], &koordy[i]);
    }
    zap(n,rast,koordx,koordy);
    for(int i = 0; i <=n; ++i){
        for(int j = i; j <= n-1; ++j) {
            if (i != j) {
                printf( format: "From %d to %d: %d ", i + 1, j + 1, rast[i][j]);
            }
        }
        printf( format: "\n");
    }
    return 0;
}

```

**Вывод:** в ходе лабораторной работы были изучены и созданы самописные функции для выполнения поставленных задач.