**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**

**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и

Автоматизированных систем

Лабораторная работа № 3.4

дисциплина: Дискретная математика

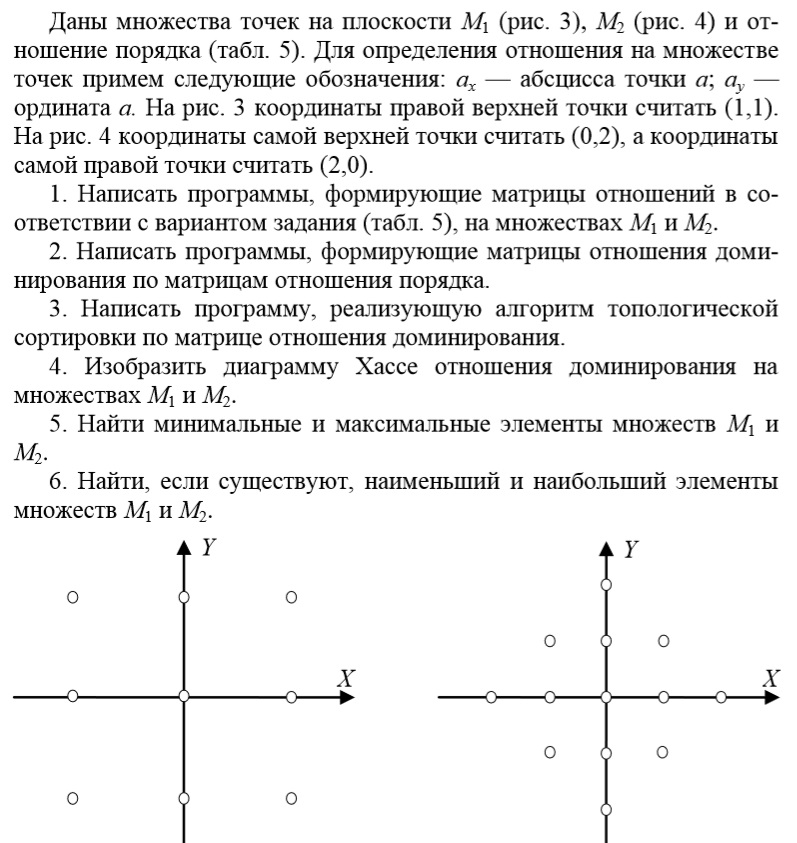
тема: **«Упорядоченные множества»**

Выполнил ст. группы

Проверил: Рязанов Ю.Д.

**Цель работы**: изучить упорядоченные множества, алгоритм топологической сортировки, научиться представлять множества диаграммами Хассе, находить минимальные (максимальные) и наименьшие (наибольшие) элементы упорядоченного множества.

**Задания**



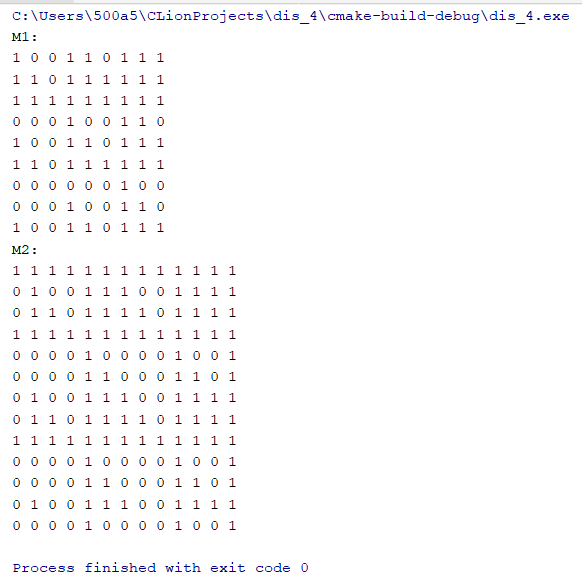
**Вариант 4**



1. Написать программы, формирующие матрицы отношений в соответствии с вариантом задания, на множествах М1 и М2.

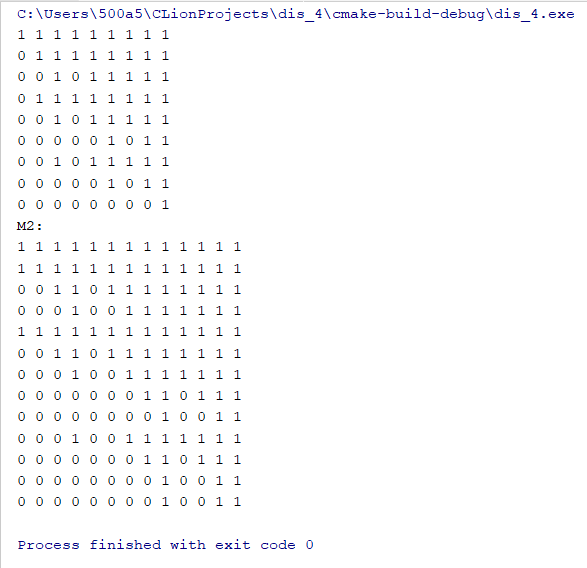
**void** mn\_T1(Point \*M1)  
{  
 **int** k = 0;  
 **for** (**int** i = -1; i < 2; i++)  
 **for** (**int** j = -1; j < 2; j++)  
 {  
 M1[k].x = i;  
 M1[k].y = j;  
 k++;  
 }  
}  
  
**void** mn\_T2(Point \*M2)  
{  
 **int** k = 0;  
 **for** (**int** i = -2; i < 3; i++)  
 **for** (**int** j = abs(i) - 2; j <= abs(abs(i) -2); j++)  
 {  
 M2[k].x = i;  
 M2[k].y = j;  
 k++;  
 }  
}

**void** get\_matr(**int** \*\*a, Point \*M, **int** n)  
{  
 **for** (**int** i = 0; i < n; i++)  
 **for** (**int** j = 0; j < n; j++)  
 **if**((M[i].x - M[i].y) <= (M[j].x - M[j].y)) {  
 a[i][j] = 1;  
 }  
 **else** {  
 a[i][j] = 0;  
 }  
}



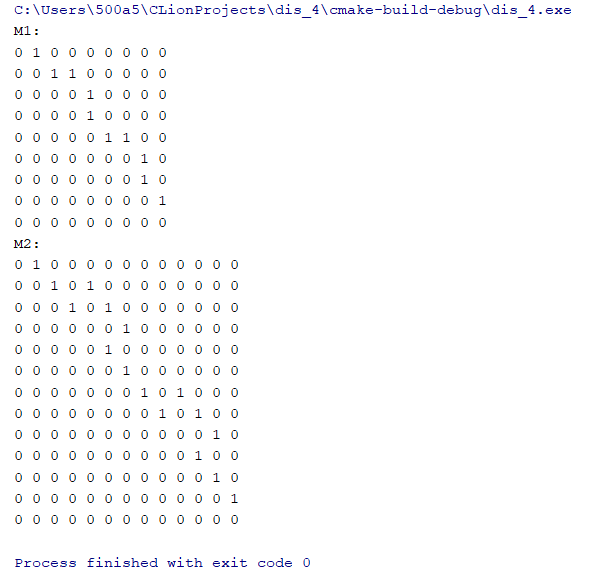
dis mat 4 image 1

**void** get\_matr(**int** \*\*a, Point \*M, **int** n)  
{  
 **for** (**int** i = 0; i < n; i++)  
 **for** (**int** j = 0; j < n; j++)  
 **if**((M[i].x - M[j].x) <= (M[j].y - M[i].y)) {  
 a[i][j] = 1;  
 }  
 **else** {  
 a[i][j] = 0;  
 }  
}



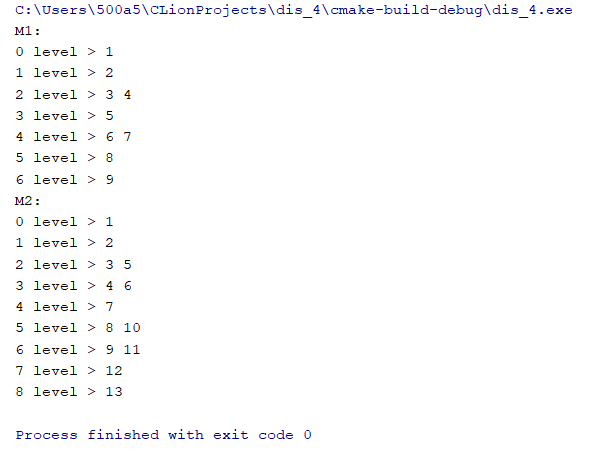
1. Написать программы, формирующие матрицы отношения доминирования по матрицам отношения порядка

**void** dominations(**int** \*\*a, **int** \*\*b, **int** n)  
{  
 **int** t, z, i, j;  
  
 **for** (j = 0; j < n; j++)  
 **for** (i = 0; i < j; i++)  
 **if** (a[i][j] == 1)  
 {  
 t = 0;  
 **for** (z = i + 1; (z < j && t == 0); z++)  
 t = a[i][z] && a[z][j];  
 b[i][j] = !t;  
 }  
}

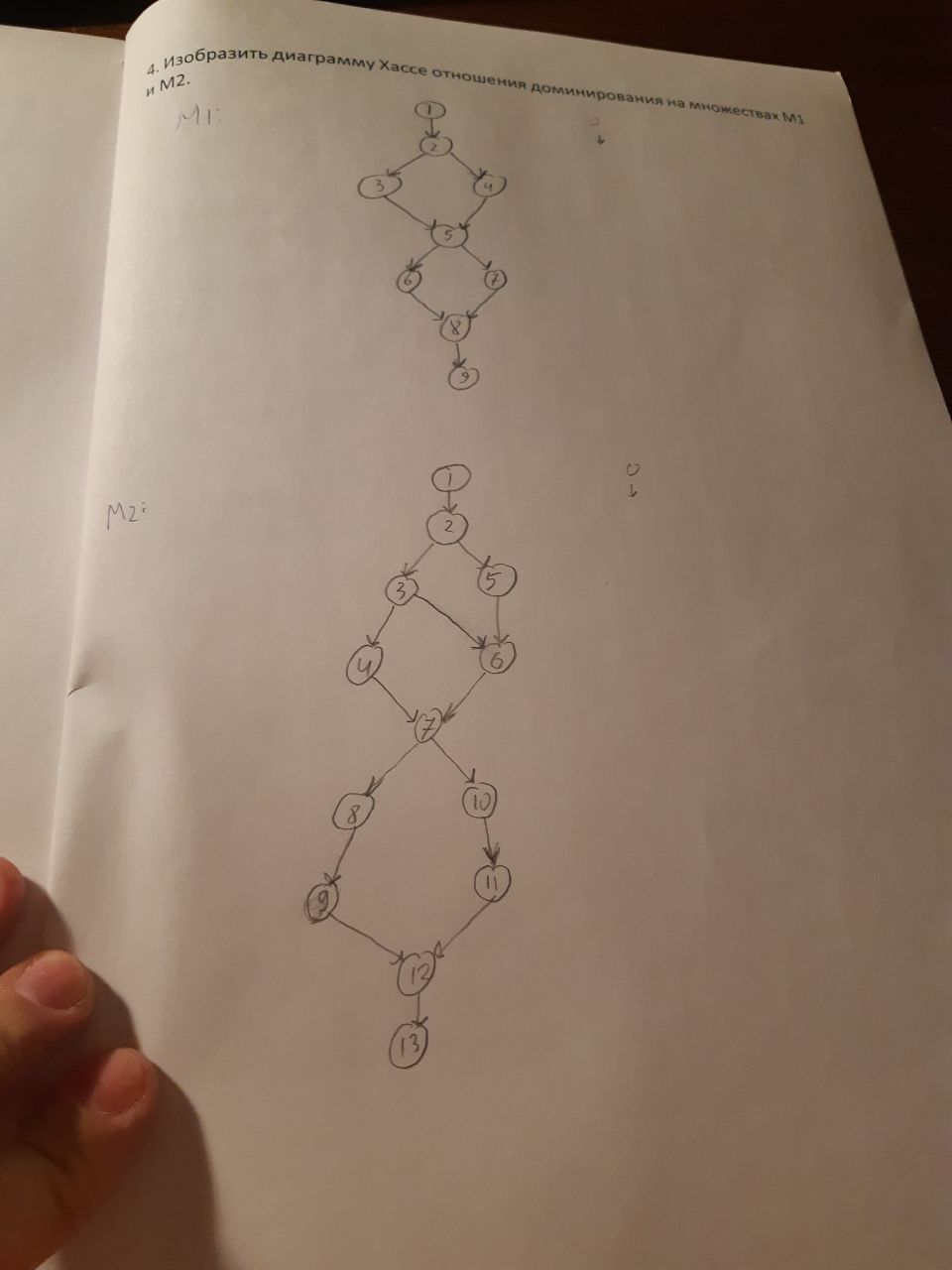


3. Написать программу, реализующую алгоритм топологической сортировки по матрице отношения доминирования.

**int** check\_otr(**int** \*a, **int** n)  
{  
 **for** (**int** i = 0; i < n; i++)  
 **if** (a[i] >= 0)  
 **return** 0;  
 **return** 1;  
}  
  
**void** sort(**int** \*\*a, **int** n)  
{  
 **int** \*w = calloc(n,**sizeof**(**int**));  
 **for**(**int** i = 0; i < n; i++)  
 **for** (**int** j = 0; j < n; j++)  
 w[i] += a[j][i];  
  
 **int** k = 0;  
 **while** (!check\_otr(w,n))  
 {  
 printf(**"%i level > "**, k);  
 **for** (**int** i = 0; i < n; i++)  
 **if**(w[i] == 0)  
 {  
 printf(**"%i "**,i+1);  
 w[i] = -(k + 1);  
 }  
 **for** (**int** i = 0; i < n; i++)  
 **if** (w[i] == -(k+1))  
 **for** (**int** j = 0; j < n; j++)  
 w[j] -= a[i][j];  
 k++;  
 printf(**"\n"**);  
 }  
}



4. Изобразить диаграмму Хассе отношения доминирования на множествах М1 и М2.



5. Найти минимальные и максимальные элементы множеств М1 и М2.

**M1**

Минимальные: 1

Максимальные: 8

**M2**

Минимальные: 1

Максимальные: 12

6. Найти, если существуют, наименьший и наибольший элементы множеств М1 и М2.

**M1**

Наименьший: 1

Наибольший: 8

**M2**

Наименьший: 1

Наибольший: 12