

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

"Белгородский государственный технологический университет им. В. Г.
Шухова"
(БГТУ им. В.Г. Шухова)

Институт энергетики, информационных технологий и управляющих
систем

Кафедра программного обеспечения вычислительной техники
и автоматизированных систем

Лабораторная работа № 3.4
по дисциплине дискретная математика
тема: Упорядоченные множества

Выполнил: студент группы ПВ-223

Игнатьев Артур Олегович

Проверил: доцент

Рязанов Юрий Дмитриевич

Белгород 2023

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

Вариант 10

$$A = \{(a, b) \mid a_x - b_x \leq b_y - a_y\}$$

Задания

Даны множества точек на плоскости M1 (рис. 3.23), M2 (рис. 3.24) и отношение порядка (табл. 3.5). Для определения отношения на множестве точек примем следующие обозначения: a_x — абсцисса точки a ; a_y — ордината a . На рис. 3.23 координаты правой верхней точки считать (1,1). На рис. 3.24 координаты самой верхней точки считать (0,2), а координаты самой правой точки считать (2,0).

1. Написать программы, формирующие матрицы отношения порядка, в соответствии с вариантом задания (табл. 3.5), на множествах M1 и M2.

```
void formR(int **m, t_struct *a, int n) {
    int i, j;
    for (i = 0; i < n; i++)
        for (j = 0; j < n; j++)
            if ((a[i].x - a[j].x) < (a[j].y - a[i].y))
                m[i][j] = 1;
}
```

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

```
void dom(int **a, int n) {
    int x, y, z;
    for (x = 0; x < n; x++)
        for (y = 0; y < n; y++)
            for (z = x + 1; z < y; z++) {
                if (a[x][z] && a[z][y])
                    a[x][y] = 0;
            }
}
```

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

```
void topsort(int **m, int *w, int n) {
    int x, y, lvl = 0;
    for (x = 0; x < n; x++) {
        w[x] = 0;
        for (y = 0; y < n; y++)
            w[x] += m[x][y];
    }
    while (empty(w, n)) {
        printf("\nУровень %d: { ", lvl);
        lvl++;
        for (x = 0; x < n; x++)
            if (w[x] == 0) {
                printf("%i ", x + 1);
                w[x] = -1;
            }
        printf("}\n");
        for (x = 0; x < n; x++)
            if (w[x] == lvl)
                w[x] = 0;
    }
}

int empty(int *w, int n) {
    int i = 0;
    while (i < n && w[i] < 0)
        i++;
    return (i < n);
}
```

4. Изобразить диаграмму Хассе отношения доминирования на множествах M1 и M2.

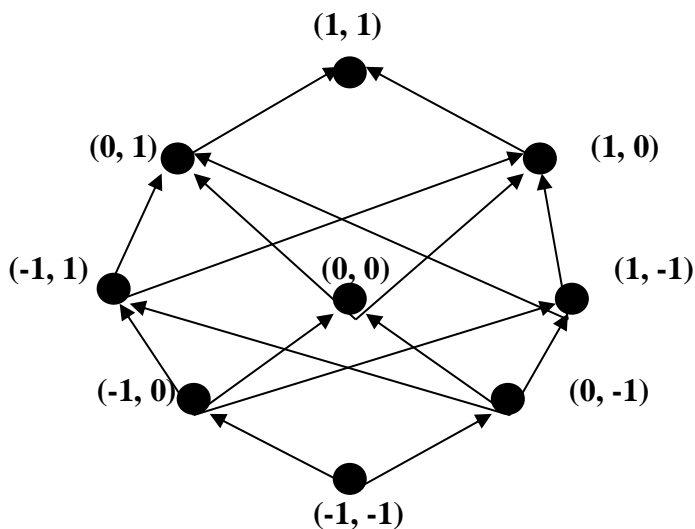


Диаграмма Хассе отношения доминирования на множестве M1

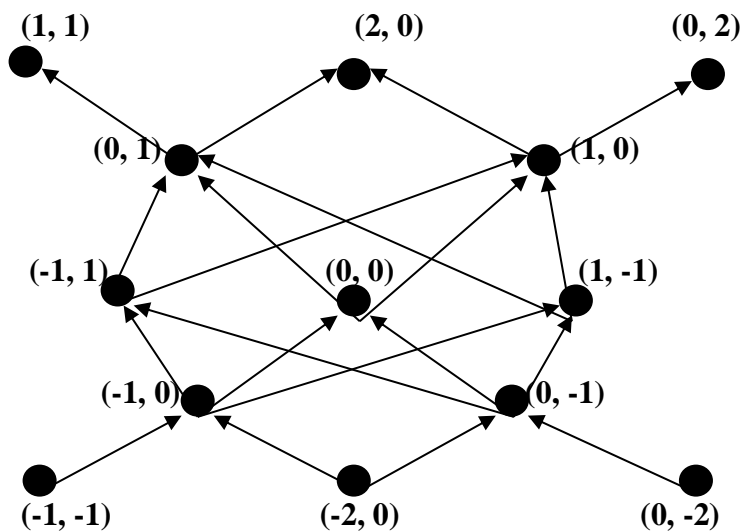


Диаграмма Хассе отношения доминирования на множестве M_2

5. Найти минимальные и максимальные элементы множеств M_1 и M_2 .

	Максимальные элементы	Минимальные элементы
M_1	(1,1)	(-1,-1)
M_2	(0,2), (1,1), (2,0)	(0,-2), (-1,-1), (-2,0)

6. Найти, если существуют, наименьший и наибольший элементы множеств M_1 и M_2 .

	Наибольший элемент	Наименьший элемент
M_1	(1,1)	(-1,-1)
M_2	Не существует	Не существует

Матрица отношения на множестве M_1 :

```

0 1 1 1 1 1 1 1 1
0 0 1 0 1 1 1 1 1
0 0 0 0 0 1 0 1 1
0 0 1 0 1 1 1 1 1
0 0 0 0 0 1 0 1 1
0 0 0 0 0 0 0 0 1
0 0 0 0 0 1 0 1 1
0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0

```

множестве M_1 :

0	1	0	1	0	0	0	0	0
0	0	1	0	1	0	1	0	0
0	0	0	0	0	1	0	1	0
0	0	1	0	1	0	1	0	0
0	0	0	0	0	1	0	1	0
0	0	0	0	0	0	0	0	1
0	0	0	0	0	1	0	1	0
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0

Матрица отношения на множестве М2:

[illegible]

Матрица отношения доминирования по матрице отношения порядка на множестве M2:

0	1	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0	0	0	0	0	0
0	0	0	0	0	1	0	1	0	0	0	0	0
0	0	1	0	1	0	1	0	0	0	0	0	0
0	0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	1	1
0	0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	1	0	0	1	1
0	1	1	1	1	1	1	1	1	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0

Вывод: на этой лабораторной работе я изучил упорядоченные множества, алгоритм топологической сортировки, научился представлять множества диаграммами Хассе, находить минимальные (максимальные) и наименьшие (наибольшие) элементы упорядоченного множества.