МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федерально автономное бюджетное образовательное учреждение высшего образования

«Севастопольский государственный университет»

кафедра Информационных систем

Куркчи Ариф Эрнестович

Институт информационных технологий и управления в технических системах

курс 3 группа ИС/б-31-о

09.03.02 Информационные системы и технологии (уровень бакалавриата)

ОТЧЕТ

по лабораторной работе №1

по дисциплине «Теория принятия решений»

на тему «Исследование применения аппарата бинарных отношений для решения задачи выбора альтернатив»

Отметка о зачете \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_

(дата)

Руководитель практикума

ст. преподаватель   А. Ю. Дрозин

(должность) (подпись) (инициалы, фамилия)

Севастополь 2016

1. ЦЕЛЬ РАБОТЫ

Исследовать применение аппарата бинарных отношений при принятии решений по выбору альтернатив.

1. ПОСТАНОВКА ЗАДАЧИ

Вариант №3

Выполнить разработку программы, реализующей определение упорядоченного множества решенийMaxR для множества *Х*, руководствуясь заданной формой графа отношений. При разработке программы использовать приведенные в теоретическом введении правила формирования множества MaxR с учетом рассмотрения вершин-приемников на каждом шаге алгоритма (задача, обратная рассматриваемой для Варианта 2). При формировании упорядоченного множества решений указывать номер яруса, на котором находятся решения. Определить эффективные решения. При разработке программы использовать вид графа отношений между решениями множества *Х*, аналогичный варианту 2. Реализовать определение эффективных решений для графов.

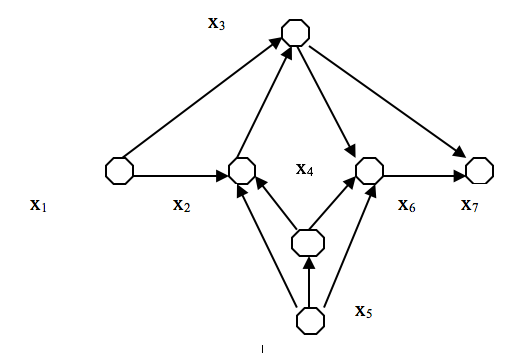


Рисунок 1 – Граф 1

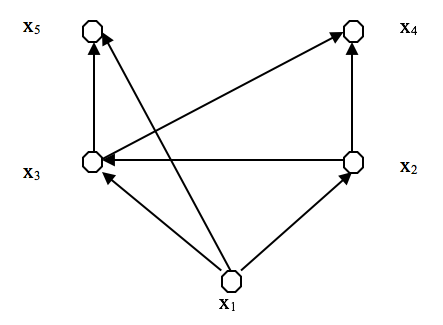


Рисунок 2 – Граф 2

1. ХОД РАБОТЫ
   1. Аналитическое решение

Приведем матрицу инцидентности графа 1:

Упорядоченное множество решений будет иметь вид:

Таблица 1 — упорядоченное множество решений для графа 1

|  |  |
| --- | --- |
| Ярус | Эффективное решение |
| 1 | X7 |
| 2 | X6 |
| 3 | X3 |
| 4 | X2 |
| 5 | X1, X5 |
| 6 | X5 |

Приведем матрицу инцидентности графа 2:

Упорядоченное множество решений для графа 2 будет иметь вид:

Таблица 2 — упорядоченное множество решений для графа 1

|  |  |
| --- | --- |
| Ярус | Эффективное решение |
| 1 | X4, x5 |
| 2 | X3 |
| 3 | X2 |
| 4 | X1 |

* 1. Программное решение

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | 1 | #include <iostream> | | 2 | #include <fstream> | | 3 | #include <vector> | | 4 |  | | 5 | using namespace std; | | 6 |  | | 7 | ifstream is("input.txt"); | | 8 | ofstream os("output.txt"); | | 9 | *int* k = 0; | | 10 |  | | 11 | *void* dot(*bool* \*\**m*, *int* *n*) { | | 12 | *char* \*str = new *char*[64]; | | 13 | sprintf(str, "graph-%d.dot", k); | | 14 | ofstream dotos(str); | | 15 | dotos << "digraph \"Ярус " << k << "\" {" << endl; | | 16 | *bool* isEmpty = true; | | 17 | for (*int* i = 1; i <= n; i++) { | | 18 | for (*int* j = 1; j <= n; j++) { | | 19 | if(m[i][j] && i != j) { | | 20 | dotos << "\t\"x" << i << "\"" << " -> " << "\"x" << j << "\"" << endl; | | 21 | isEmpty = false; | | 22 | } | | 23 | } | | 24 | } | | 25 | dotos << "}" << endl; | | 26 | dotos.close(); | | 27 |  | | 28 | if(!isEmpty) { | | 29 | sprintf(str, "dot graph-%d.dot -Tpng -O", k); | | 30 | system(str); | | 31 | } | | 32 | sprintf(str, "rm -f graph-%d.dot", k); | | 33 | system(str); | | 34 | k++; | | 35 | } | | 36 |  | | 37 | *void* draw(*bool* \*\**m*, *int* *n*) { | | 38 | dot(m,n); | | 39 | os << "== MATRIX ==" << endl; | | 40 | for (*int* i = 1; i <= n; i++) { | | 41 | for (*int* j = 1; j <= n; j++) { | | 42 | if ((i == j && m[i][j]) || (m[i][i] && m[j][j])) { | | 43 | os << "╋"; | | 44 | } else if (m[i][i]) { | | 45 | os << "━"; | | 46 | } else if (m[j][j]) { | | 47 | os << "┃"; | | 48 | } else { | | 49 | if (m[i][j]) { | | 50 | os << 1; | | 51 | } else { | | 52 | os << 0; | | 53 | } | | 54 | } | | 55 | } | | 56 | os << endl; | | 57 | } | | 58 | os << "============" << endl; | | 59 | } | | 60 |  | | 61 | *int* main() { | | 62 |  | | 63 | *char* dir; | | 64 | *int* n, k; | | 65 | is >> n >> k >> dir; | | 66 |  | | 67 | *bool* \*\*m = new *bool* \*[n + 1]; | | 68 | for (*int* i = 0; i <= n; i++) { | | 69 | m[i] = new *bool*[n + 1]; | | 70 | } | | 71 | memset(m, 0, sizeof(m)); | | 72 |  | | 73 | *int* a, b; | | 74 | for (*int* i = 0; i < k; i++) { | | 75 | is >> a >> b; | | 76 | m[a][b] = true; | | 77 | } | | 78 | os << "Initial" << endl; | | 79 | draw(m, n); | | 80 |  | | 81 | *bool* isEnded = false; | | 82 | vector<*int*> level; | | 83 | *int* lvl = 1; | | 84 | while (!isEnded) { | | 85 | level.clear(); | | 86 | for (*int* i = 1; i <= n; i++) { | | 87 | if (m[i][i]) { | | 88 | continue; | | 89 | } | | 90 | *bool* isEmpty = true; | | 91 | for (*int* j = 1; j <= n && isEmpty; j++) { | | 92 | if ((dir == 'd' && m[i][j] && !m[i][i] && !m[j][j]) || | | 93 | (dir == 'u' && m[j][i])) { | | 94 | isEmpty = false; | | 95 | } | | 96 | } | | 97 | if (isEmpty) { | | 98 | level.push\_back(i); | | 99 | } | | 100 | } | | 101 | if (level.size()) { | | 102 | os << "Level " << lvl++ << ": "; | | 103 | for (vector<*int*>::iterator it = level.begin(); it != level.end(); it++) { | | 104 | for (*int* i = 1; i <= n; i++) { | | 105 | m[\*it][i] = false; | | 106 | } | | 107 | m[\*it][\*it] = true; | | 108 | os << \*it << " "; | | 109 | } | | 110 | os << endl; | | 111 | draw(m, n); | | 112 | } else { | | 113 | isEnded = true; | | 114 | } | | 115 | } | | 116 |  | | 117 | return 0; | | 118 | } | |  |

ВЫВОДЫ

В ходе лабораторной работы были исследованы бинарные отношения задач принятия решения. Были решены задач принятия решения с использованием выбора эффективных решений, основанных на отношении предпочтения, при котором на верхний ярус выносятся наиболее эффективные решения. Написана программа, реализующая генерацию отношения предпочтения и, следовательно, множества MaxR как на базе вершин-стоков, так и на базе вершин-истоков. Так же программа на каждом шагу генерирует соответствующую матрицу и текущую версию графа в виде PNG изображения, используя средство визуализации GraphViz.