МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Севастопольский государственный университет»

кафедра Информационные системы

Сирота Марина Романовна

Институт информационных технологий и управления в технических системах

курс 3 группа ИС/б-32-о

09.03.02 Информационные системы и технологии (уровень бакалавриата)

ОТЧЕТ

по лабораторной работе №2

по дисциплине: «Теория принятия решений»

по теме: «Исследование применения аппарата теории полезности для описания бинарных отношений при принятии решений»

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

(дата)

Руководитель практикума

доц. Кротов К.В.

(должность) (подпись) (инициалы, фамилия)

Севастополь

2018

1. ЦЕЛЬ РАБОТЫ

Исследовать применение аппарата теории полезности при принятии решений по выбору альтернатив.

1. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Вариант 1. Используя метод, реализующий формирование множеств  и , а также их последующий анализ; выполнить для заданного вида матрицы отношения предпочтения *А1* определение значений функции полезности *U()* решений и определение по формируемым значениям функции полезности эффективных решений  ∈ *X* . Матрица отношения предпочтения имеет следующий вид:

1. ВЫПОЛНЕНИЕ РАБОТЫ
   1. Аналитическое решение
2. , ;
3. :

 

;

1. :

 

;

1. :

 

;

1. :

 

;

1. :

 ; 

;

1. :

 ; 

.

Эффективным решение является решение .

* 1. Код программы на C++

#include <iostream>

using namespace std;

const int n = 7;

void search(int mas[n][n]) {

setlocale(LC\_ALL, "RUS");

float U[n];

int XPlus[n];

int XMinus [n];

int k = 1;

bool XPlusEmpty = true, XMinusEmpty = true, Inter = false;

int min = -1, max = -1;

int maxU = 0;

U[0] = 0.0f;

for (int p = 0; p < n - 1; p++)

{

XPlusEmpty = true;

XMinusEmpty = true;

Inter = false;

for (int i = 0; i < n ; i++)

{

XPlus[i] = 0;

XMinus[i] = 0;

}

for (int i = 0; i < k; i++)

{

if (mas[i][p + 1] == 1)

{

XPlus[i] = 1;

XPlusEmpty = false;

}

if (mas[p + 1][i] == 1)

{

XMinus[i] = 1;

XMinusEmpty = false;

}

}

for (int j = 0; j < n ; j++)

if (XPlus[j] == 1 && XMinus[j] == 1)

{

Inter = true;

}

if (XPlusEmpty == true && XMinusEmpty == false)

U[p + 1] = U[p] + 1.0f;

else if (XMinusEmpty == true && XPlusEmpty == false)

U[p + 1] = U[p] - 1.0f;

else if (XMinusEmpty == false && XPlusEmpty == false && Inter == true)

{

for (int i = 0; i <= k; i++)

{

if (XMinus[i] == 1 && XPlus[i] == 1) U[p + 1] = U[i];

}

}

else if (XMinusEmpty == false && XPlusEmpty == false && Inter == false)

{

for (int z = 0; z <= k; z++)

{

if (XPlus[z] == 1)

{

min = z;

for (int i = 0; i <= k; i++)

if (mas[i][z] != 1 && XPlus[i] == 1) break;

}

}

for (int z = 0; z <= k; z++)

{

if (XMinus[z] == 1)

{

max = z;

for (int i = 0; i <= k; i++)

if (mas[z][i] != 1 && XMinus[i] == 1)

{

max = z;

break;

}

}

}

U[p + 1] = (U[min] + U[max]) / 2.0f;

}

k++;

cout << "X" << (p + 1) << "- : ";

for (int i = 0; i < n; i++)

{

if (XMinus[i] == 1)

cout << "X" << i+1 << " ";

}

cout << endl;

cout << "X" << (p + 1) << "+ : ";

for (int i = 0; i < n; i++)

{

if (XPlus[i] == 1)

cout << "X" << i+1 << " ";

}

cout << endl;

cout << endl;

}

cout << "U: ";

for (int i = 0; i < n; i++)

{

cout << U[i] << " ";

if (U[i] > U[maxU])

maxU = i;

}

cout << endl;

cout << "Эффективные решения : ";

for (int i = 0; i < n; i++)

if (U[i] == U[maxU])

cout << "X" << (i + 1) << " ";

}

int main()

{

const int n = 7;

int A[n][n]={

{ 0, 0, 0, 0, 0, 0, 0 },

{ 1, 0, 1, 0, 0, 0, 0 },

{ 0, 0, 0, 1, 0, 0, 1 },

{ 0, 1, 0, 0, 1, 1, 0 },

{ 0, 1, 0, 0, 0, 0, 0 },

{ 1, 0, 0, 1, 0, 0, 0 },

{ 0, 0, 1, 0, 0, 1, 0 },

};

search(A);

cout << endl;

system("pause");

}

* 1. Результаты выполнения программы

Рисунок 3.1 ‒ Результаты для графа 1

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.

Была написана программа, реализующая генерацию отношения предпочтения и множества MaxR как на базе вершин-приемников.