Отчёт к лабораторной работе

по дисциплине  
«Теория принятия решений»

выполнил   
студент гр. ИС/б-18-1-з Демиденко А. А.  
принял Кротов К. В.

Лабораторной работа № 1  
«Исследование применения аппарата бинарных отношений для решения задачи выбора альтернатив»

## Цель работы

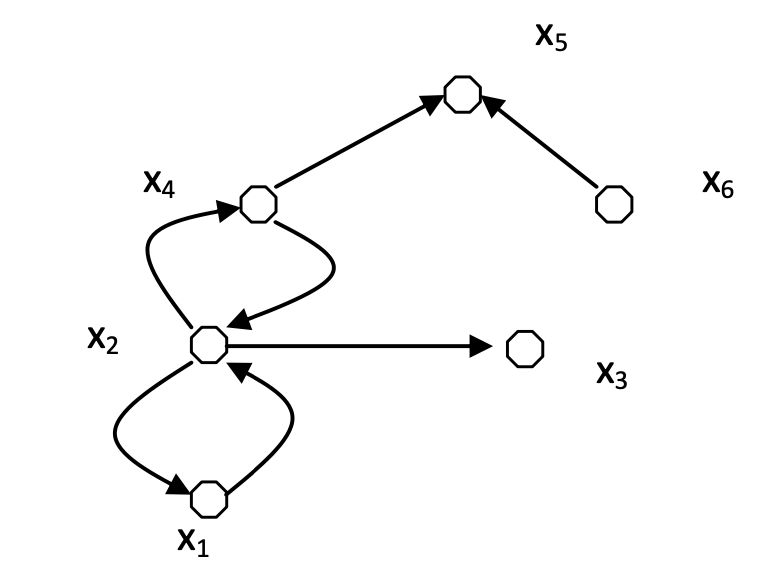
Исследовать применение аппарата бинарных отношений при принятии решений по выбору альтернатив.

## Программа выполнения работы

1. Для **Варианта 1** задания на работу, связанного с формированием подмножества максимальных элементов MaxR множества Х, необходимо по заданному варианту графа отношений предпочтения между решениями сформировать матрицу А отношения R (где R– отношение ≻). При этом убедиться, что первый элемент множества Х является строго независящим от других решений.
2. Выполнить формирование множества MaxR вручную для заданного вида графа и соответствующего ему вида матрицы А.
3. Выполнить формирование программного кода соответствующей процедуры определения множества MaxR, при этом возможно руководствоваться ориентировочным видом процедуры определения этого множества, предложенным в теоретическом введении данной лабораторной работы.
4. Выполнить вывод результатов работы процедуры и сравнить полученные в процедуре результаты с результатами, сформированными аналитически.
5. Изменить исходные данные программы, используя графы отношений из примера 5 (Рис 7). Проверить получаемые с использованием процедуры результаты с аналитическим результатами, формируемыми для этих графов.

## Ход работы

Сформируем матрицу отношения ≻ для графа отношений, соответствующего варианту № 1, изображенного на рисунке 1.

  
Рисунок 1 – Граф отношений варианта №1

Матрица отношений выглядит следующим образом:

Рассмотрим каждую вершину графа, изображённого на рисунке 1 на предмет включения в множество MaxRX:

* Решение x1 не является доминируемым, оно эквивалентно решениям x2 и x4, при этом ни одно решение не доминирует над ними. Из этого следует, что решения x1, x2 и x4 должны быть включены во множество MaxRX.
* Решение x3 доминируемо решением x2, в множество MaxRX не включаем.
* Решение x5 доминируемо решениями x4 и x6, в множество MaxRX не включаем.
* Решение x6 никаким решением не доминируемо и ни с каким решением не эквивалентно, включим его в множество MaxRX.

Итоговое множество MaxRX состоит из следующих элементов: x1, x2, x4 и x6.

Далее, реализуем программу для формирования множества MaxRX:

import Control.Monad (guard)

type Row a = [a]

type Matrix a = [Row a]

type RowI a = [(Int, a)]

type MatrixI a = [(Int, RowI a)]

addIndexes :: Matrix a -> MatrixI a

addIndexes matrix = zip [1..] $ zip [1..] <$> matrix

withRelation :: MatrixI Bool -> RowI Int

withRelation matrix = do

(i, row) <- matrix

(j, column) <- row

guard column

pure (i, j)

replace :: Int -> a -> [a] -> [a]

replace \_ \_ [] = []

replace 0 val (\_:xs) = val : xs

replace n val (x:xs) = x : replace (n - 1) val xs

get :: Row a -> Int -> a

get s i = s !! (i - 1)

get2 :: Matrix a -> Int -> Int -> a

get2 s i = get (get s i)

getMaxR :: Matrix Bool -> Row Bool

getMaxR matrix = foldr selector maxR . withRelation . addIndexes $ matrix

where

maxR = replicate (length matrix) True

selector (i, j) res

| not $ get2 matrix j i = replace (j - 1) False res

| otherwise = if not $ get res i then replace (j - 1) False res else res

Множество MaxRX, сформированное при помощи программы совпадает с полученным вручную:

*-- >>> getMaxR input*

*-- [True,True,False,True,False,True]*

input :: Matrix Bool

input =

[ [ False, True, False, False, False, False ]

, [ True, False, True, True, False, False ]

, [ False, False, False, False, False, False ]

, [ False, True, False, False, True, False ]

, [ False, False, False, False, False, False ]

, [ False, False, False, False, True, False ]

]

Теперь проверим работу программы на данных из рисунка 7 методички:

*-- >>> getMaxR input7a*

*-- [True,True,False,False,True]*

input7a :: Matrix Bool

input7a =

[ [ False, False, False, True, False ]

, [ False, False, True, False, False ]

, [ False, False, False, True, True ]

, [ False, False, True, False, False ]

, [ False, False, True, False, False ]

]

*-- >>> getMaxR input7b*

*-- [True,True,False,False,False]*

input7b :: Matrix Bool

input7b =

[ [ False, True, False, True, False ]

, [ True, False, True, False, False ]

, [ False, False, False, True, True ]

, [ False, False, True, False, False ]

, [ False, False, False, False, False ]

]

Все результаты совпадают с аналитическими.

## Выводы

В ходе лабораторной работы была рассмотрена задача выбора оптимальных решений из альтернатив при помощи аппарата бинарных отношений. В частности, была рассмотрена ситуация, когда для списка альтернатив не может быть сформировано множество предпочитаемых решений.

Для решения задачи было сформировано множество MaxRX максимальных решений по следующим правилам:

* решение исключается из MaxRX если оно доминируемо другим решением;
* решение исключается из MaxRX если оно эквивалентно другому решению, которое не включено в MaxRX;
* если решение никем не доминируемо, то оно включается в MaxRX.

В результате максимальные множества, полученные аналитически и программно, совпали.