# Лабораторная работа № 3.2 Транзитивное замыкание отношения

|  |  |
| --- | --- |
| Цель работы: | изучить и выполнить сравнительный анализ алгоритмов вычисления транзитивного замыкания отношения. |

## Задания

1. Изучить и программно реализовать алгоритмы объединения степеней и Уоршалла для вычисления транзитивного замыкания отношения.
2. Разработать и программно реализовать генератор отношения на множестве и содержащих заданное число пар.
3. Разработать и написать программу, которая генерирует 1000 отношений на множестве мощности с заданным числом пар, для каждого отношения вычисляет транзитивное замыкание двумя алгоритмами и определяет количество итераций самого вложенного цикла для каждого алгоритма. Это количество для различных отношений на множестве мощности с заданным числом пар может быть разным, поэтому программа должна также определять минимальное и максимальное из них. Выполнить программу при . Результат для каждого представить в виде таблицы.

## Описание подпрограмм

1. Спецификация функции transitivityClosureSlow
   1. Заголовок: procedure transitivityClosureSlow(A: boolVariety; var C: boolVariety; var counter: longint);
   2. Назначение: записывает в C результат выполнения транзитивного замыкания отношения A при помощи алгоритма объединения квадратов (counter — количество итераций самого вложенного цикла);
   3. Входные параметры: A
   4. Выходные параметры: C, counter
2. Спецификация функции transitivityClosure
   1. Заголовок: procedure transitivityClosure(A: boolVariety; var C: boolVariety; var counter: longint);
   2. Назначение: записывает в C результат выполнения транзитивного замыкания отношения A при помощи алгоритма объединения степеней (counter — количество итераций самого вложенного цикла);
   3. Входные параметры: A
   4. Выходные параметры: C, counter
3. Спецификация функции transitivityClosureImproved
   1. Заголовок: procedure transitivityClosureImproved(A: boolVariety; var C: boolVariety; var counter: longint);
   2. Назначение: записывает в C результат выполнения транзитивного замыкания отношения A при помощи улучшенного алгоритма объединения степеней (counter — количество итераций самого вложенного цикла);
   3. Входные параметры: A
   4. Выходные параметры: C, counter
4. Спецификация функции transitivityClosureWarshall
   1. Заголовок: procedure transitivityClosureWarshall(A: boolVariety; var C: boolVariety; var counter: longint);
   2. Назначение: записывает в C результат выполнения транзитивного замыкания отношения A при помощи алгоритма Уоршалла степеней (counter — количество итераций самого вложенного цикла);
   3. Входные параметры: A
   4. Выходные параметры: C, counter
5. Спецификация функции transitivityClosureWarshallImproved
   1. Заголовок: procedure transitivityClosureWarshallImproved(A: boolVariety; var C: boolVariety; var counter: longint);
   2. Назначение: записывает в C результат выполнения транзитивного замыкания отношения A при помощи улучшенного алгоритма Уоршалла степеней (counter — количество итераций самого вложенного цикла);
   3. Входные параметры: A
   4. Выходные параметры: C, counter

## Таблица результатов

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N | Число пар в отношении |  | |  | |  | |  | |  | |
| min | max | min | max | min | max | min | max | min | max |
| 5 | Алгоритм объединения квадратов | 125 | 125 | 224 | 276 | 126 | 132 | 68 | 99 | 25 | 25 |
| Алгоритм объединения степеней | 375 | 375 | 276 | 348 | 153 | 208 | 103 | 123 | 75 | 75 |
| Улучшенный алгоритм объединения степеней | 125 | 125 | 276 | 348 | 143 | 208 | 73 | 123 | 25 | 25 |
| Алгоритм Уоршалла | 125 | 125 | 125 | 125 | 125 | 125 | 125 | 125 | 125 | 125 |
| Улучшенный алгоритм Уоршалла | 29 | 29 | 53 | 65 | 89 | 101 | 117 | 121 | 125 | 125 |
| 10 | Алгоритм объединения квадратов | 993 | 1000 | 985 | 2054 | 482 | 663 | 268 | 530 | 100 | 100 |
| Алгоритм объединения степеней | 7944 | 8000 | 3357 | 5461 | 1332 | 2192 | 1008 | 1589 | 800 | 800 |
| Улучшенный алгоритм объединения степеней | 993 | 1000 | 2003 | 5461 | 582 | 942 | 288 | 658 | 100 | 100 |
| Алгоритм Уоршалла | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 |
| Улучшенный алгоритм Уоршалла | 109 | 109 | 469 | 640 | 766 | 955 | 883 | 973 | 1000 | 1000 |
| 15 | Алгоритм объединения квадратов | 3367 | 3375 | 2659 | 5369 | 1017 | 1936 | 609 | 823 | 225 | 225 |
| Алгоритм объединения степеней | 43771 | 43875 | 9762 | 20083 | 4502 | 8558 | 3332 | 5998 | 2925 | 2925 |
| Улучшенный алгоритм объединения степеней | 3367 | 3375 | 4773 | 8897 | 1227 | 2354 | 669 | 1048 | 225 | 225 |
| Алгоритм Уоршалла | 3375 | 3375 | 3375 | 3375 | 3375 | 3375 | 3375 | 3375 | 3375 | 3375 |
| Улучшенный алгоритм Уоршалла | 239 | 239 | 1765 | 2759 | 2815 | 3179 | 3151 | 3319 | 3375 | 3375 |