Белорусский Государственный Университет Информатики и Радиоэлектроники

**Отчет по Лабораторной работе №4**

Выполнение операций над соответствиями

Работу выполнили:

Локтев К. А., Дюбайло М. А., Виноградова П. С. и Макаревич К.С.

Группа 021702

Проверила:

Гулякина Н. А.

***Постановка задачи:***

Даны 2 соответствия. Выполнить операции над соответствиями: объединение, пересечение, разность, симметрическая разность, инверсия, композиция, сужение и продолжение.

***Уточнение постановки задачи:***

Области отправления, области прибытия, графики соответствий, множества сужения и графики продолжения задаются пользователем методом перечисления.

Элементами областей отправления, областей прибытия, пар графиков и множества сужения соответствий могут быть любые натуральные числа.

Мощность каждой области отправления и каждой области прибытия соответствий, а также множества сужения, задаётся пользователем и может быть натуральным числом от 1 до 30.

Мощность каждого графика соответствия может быть натуральным числом от 1 до 900.

***Определения:***

Множество – совокупность некоторых элементов, мыслимых, как единое целое.

Кортеж – упорядоченный набор некоторых элементов.

Мощность множества – количество элементов множества.

Длина кортежа – количество элементов в кортеже.

График – множество, элементами которого являются кортежи длины 2 (пары, упорядоченные пары).

Соответствием между множествами и называется тройка или , где – область отправления соответствия, – область прибытия соответствия, а - произвольное подмножество графика .

Объединение множеств – множество, содержащее все элементы исходных множеств. Пусть даны множества и , тогда = {}.

Пересечение множеств – множество, которому принадлежат все элементы, принадлежащие одновременно всем исходным множествам. Пусть даны множества и , тогда .

Разность двух множеств – множество, содержащее все элементы первого множества, не принадлежащие второму множеству. Пусть даны множества и , тогда .

Симметрическая разность двух множеств – множество, содержащее все элементы первого множества, не принадлежащие второму множеству, и все элементы второго множества, не принадлежащие первому множеству. Пусть даны множества и , тогда = .

Инверсия графика – график, во всех упорядоченных парах которого порядок изменён на противоположный относительно исходного графика.

Пусть дан график , тогда .

Композиция двух графиков – график пар таких, что пара принадлежит первому графику, а пара - второму графику.

Пусть даны графики и , тогда .

Объединение двух соответствий - соответствие, область отправления которого – результат объединения областей отправления исходных соответствий, область прибытия – результат объединения областей прибытия исходных соответствий, а график – результат объединения графиков исходных соответствий.

Пусть даны соответствия и ,

тогда .

Пересечение двух соответствий - соответствие, область отправления которого – результат пересечения областей отправления исходных соответствий, область прибытия – результат пересечения областей прибытия исходных соответствий, а график – результат пересечения графиков исходных соответствий.

Пусть даны соответствия и ,

тогда .

Разность двух соответствий - соответствие, область отправления которого – результат разности областей отправления исходных соответствий, область прибытия – результат разности областей прибытия исходных соответствий, а график – результат разности графиков исходных соответствий.

Пусть даны соответствия и ,

тогда .

Симметрическая разность двух соответствий - соответствие, область отправления которого – результат симметрической разности областей отправления исходных соответствий, область прибытия – результат симметрической разности областей прибытия исходных соответствий, а график – результат симметрической разности графиков исходных соответствий.

Пусть даны соответствия и ,

тогда .

Инверсия соответствия – соответствие, область отправления которого совпадает с областью прибытия исходного соответствия, область прибытия совпадает с областью отправления исходного соответствия, а график совпадает с инверсией графика исходного соответствия.

Пусть дано соответствие , тогда .

Композиция двух соответствий – соответствие, область отправления которого совпадает с областью отправления первого соответствия, область прибытия совпадает с областью прибытия второго соответствия, а график совпадает с композицией графиков первого и второго соответствий.

Пусть даны соответствия и , тогда *.*

Сужение соответствия на множество – соответствие, график которого является результатом пересечения графика исходного соответствия и декартова произведения , причём множество является подмножеством области отправления исходного соответствия.

Пусть дано соответствие и , тогда .

Продолжение соответствия – соответствие, график которого является надмножеством графика исходного соответствия.

Пусть дано соответствие , тогда продолжением будет называться любое соответствие , где .

***Описание алгоритма:***

1. **Ввод данных**
   1. Создаётся пустое множество, которое ляжет в основу области отправления первого соответствия (далее - )
   2. Пользователь вводит мощность множества (далее - )
   3. Задаётся число i, равное 0
   4. Число i увеличивается на 1
   5. Пользователь вводит i-тый элемент множества
   6. Если число i меньше , алгоритм переходит к пункту 1.4
   7. Создаётся пустое множество, которое ляжет в основу области прибытия первого соответствия (далее - )
   8. Пользователь вводит мощность множества (далее - )
   9. Число i приравнивается к 0
   10. Число i увеличивается на 1
   11. Пользователь вводит i-тый элемент множества
   12. Если число i меньше , алгоритм переходит к пункту 1.10
   13. Создаётся пустой график, который ляжет в основу графика первого соответствия (далее - )
   14. Пользователь вводит мощность графика (далее - )
   15. Число i приравнивается к 0
   16. Число i увеличивается на 1
   17. Создаётся пустая пара p
   18. Пользователь вводит первый и второй элементы пары p
   19. Задаётся число j, равное 0
   20. Число j увеличивается на 1
   21. Если число j равно , то:
       1. На экран выводится сообщение о том, что первого элемента введённой пользователем пары нет в области отправления первого соответствия
       2. Алгоритм переходит к пункту 1.17
   22. Если j-тый элемент множества не равен первому элементу пары p, алгоритм переходит к пункту 1.20
   23. Число j приравнивается к 0
   24. Число j увеличивается на 1
   25. Если число j равно , то:
       1. На экран выводится сообщение о том, что второго элемента введённой пользователем пары нет в области прибытия первого соответствия
       2. Алгоритм переходит к пункту 1.17
   26. Если j-тый элемент множества не равен второму элементу пары p, алгоритм переходит к пункту 1.24
   27. Пара p записывается в график
   28. Если число i меньше , алгоритм переходит к пункту 1.16
   29. Создаётся пустое множество, которое ляжет в основу области отправления второго соответствия (далее - )
   30. Пользователь вводит мощность множества (далее - )
   31. Задаётся число i, равное 0
   32. Число i увеличивается на 1
   33. Пользователь вводит i-тый элемент множества
   34. Если число i меньше , алгоритм переходит к пункту 1.32
   35. Создаётся пустое множество, которое ляжет в основу области прибытия второго соответствия (далее - )
   36. Пользователь вводит мощность множества (далее - )
   37. Число i приравнивается к 0
   38. Число i увеличивается на 1
   39. Пользователь вводит i-тый элемент множества
   40. Если число i меньше , алгоритм переходит к пункту 1.38
   41. Создаётся пустой график, который ляжет в основу графика второго соответствия (далее - )
   42. Пользователь вводит мощность графика (далее - )
   43. Число i приравнивается к 0
   44. Число i увеличивается на 1
   45. Создаётся пустая пара p
   46. Пользователь вводит первый и второй элементы пары p
   47. Задаётся число j, равное 0
   48. Число j увеличивается на 1
   49. Если число j равно , то:
       1. На экран выводится сообщение о том, что первого элемента введённой пользователем пары нет в области отправления второго соответствия
       2. Алгоритм переходит к пункту 1.45
   50. Если j-тый элемент множества не равен первому элементу пары p, алгоритм переходит к пункту 1.20
   51. Число j приравнивается к 0
   52. Число j увеличивается на 1
   53. Если число j равно , то:
       1. На экран выводится сообщение о том, что второго элемента введённой пользователем пары нет в области прибытия второго соответствия
       2. Алгоритм переходит к пункту 1.45
   54. Если j-тый элемент множества не равен второму элементу пары p, алгоритм переходит к пункту 1.52
   55. Пара p записывается в график
   56. Если число i меньше , алгоритм переходит к пункту 1.44
   57. Множество выводится на экран
   58. Множество выводится на экран
   59. График выводится на экран
   60. Множество выводится на экран
   61. Множество выводится на экран
   62. График выводится на экран
2. **Операция объединения**
   1. Создаётся пустое множество, которое ляжет в основу области отправления результирующего соответствия (далее - )
   2. Создаётся пустое множество, которое ляжет в основу области прибытия результирующего соответствия (далее - )
   3. Создаётся пустой график, который ляжет в основу графика результирующего соответствия (далее - )
   4. Задаётся число i, равное 0
   5. Число i увеличивается на 1
   6. i-тый элемент множества записывается во множество
   7. Если число i меньше , алгоритм переходит к пункту 2.5
   8. Число i приравнивается к 0
   9. Число i увеличивается на 1
   10. Задаётся число j, равное 0
   11. Число j увеличивается на 1
   12. Если i-тый элемент множества равен j-тому элементу множества , то:
       1. Если число i равно ||, алгоритм переходит к пункту 2.15
       2. Алгоритм переходит к пункту 2.9
   13. Если число j равно , то:
       1. i-тый элемент множества записывается во множество
       2. Алгоритм переходит к пункту 2.12.1
   14. Алгоритм переходит к пункту 2.11
   15. Множество – область отправления результирующего соответствия, выводится на экран
   16. Число i приравнивается к 0
   17. Число i увеличивается на 1
   18. i-тый элемент множества записывается во множество
   19. Если число i меньше , алгоритм переходит к пункту 2.17
   20. Число i приравнивается к 0
   21. Число i увеличивается на 1
   22. Число j приравнивается к 0
   23. Число j увеличивается на 1
   24. Если i-тый элемент множества равен j-тому элементу множества , то:
       1. Если число i равно ||, алгоритм переходит к пункту 2.27
       2. Алгоритм переходит к пункту 2.21
   25. Если число j равно , то:
       1. i-тый элемент множества записывается во множество
       2. Алгоритм переходит к пункту 2.24.1
   26. Алгоритм переходит к пункту 2.22
   27. Множество – область прибытия результирующего соответствия, выводится на экран
   28. Число i приравнивается к 0
   29. Число i увеличивается на 1
   30. i-тая пара графика записывается в график
   31. Если число i меньше , алгоритм переходит к пункту 2.28
   32. Число i приравнивается к 0
   33. Число i увеличивается на 1
   34. Число j приравнивается к 0
   35. Число j увеличивается на 1
   36. Если первый элемент i-той пары графика равен первому элементу j-той пары графика , то:
       1. Если второй элемент i-той пары графика равен второму элементу j-той пары графика , то:
          1. Если число i равно , алгоритм переходит к пункту 2.39
          2. Алгоритм переходит к пункту 2.33
   37. Если число j равно , то:
       1. i-тая пара графика записывается в график
       2. Алгоритм переходит к пункту 2.36.1.1
   38. Алгоритм переходит к пункту 2.35
   39. График – график результирующего соответствия, выводится на экран
3. **Операция пересечения**
   1. Создаётся пустое множество, которое ляжет в основу области отправления результирующего соответствия (далее - )
   2. Создаётся пустое множество, которое ляжет в основу области прибытия результирующего соответствия (далее - )
   3. Создаётся пустой график, который ляжет в основу графика результирующего соответствия (далее - )
   4. Задаётся число i, равное 0
   5. Число i увеличивается на 1
   6. Задаётся число j, равное 0
   7. Число j увеличивается на 1
   8. Если i-тый элемент множества равен j-тому элементу множества , то:
      1. i-тый элемент множества записывается во множество
      2. Если число i равно ||, алгоритм переходит к пункту 3.11
      3. Алгоритм переходит к пункту 3.5
   9. Если число j равно , алгоритм переходит к пункту 3.8.2
   10. Алгоритм переходит к пункту 3.7
   11. Множество – область отправления результирующего соответствия, выводится на экран
   12. Число i приравнивается к 0
   13. Число i увеличивается на 1
   14. Число j приравнивается к 0
   15. Число j увеличивается на 1
   16. Если i-тый элемент множества равен j-тому элементу множества , то:
       1. i-тый элемент множества записывается во множество
       2. Если число i равно ||, алгоритм переходит к пункту 3.19
       3. Алгоритм переходит к пункту 3.13
   17. Если число j равно , алгоритм переходит к пункту 3.16.2
   18. Алгоритм переходит к пункту 3.15
   19. Множество – область прибытия результирующего соответствия, выводится на экран
   20. Число i приравнивается к 0
   21. Число i увеличивается на 1
   22. i-тая пара графика записывается в график
   23. Если число i меньше , алгоритм переходит к пункту 3.21
   24. Число i приравнивается к 0
   25. Число i увеличивается на 1
   26. Число j приравнивается к 0
   27. Число j увеличивается на 1
   28. Если первый элемент i-той пары графика равен первому элементу j-той пары графика , то:
       1. Если второй элемент i-той пары графика равен второму элементу j-той пары графика , то:
          1. i-тая пара графика записывается в график
          2. Если число i равно , алгоритм переходит к пункту 3.31
          3. Алгоритм переходит к пункту 3.25
   29. Если число j равно , алгоритм переходит к пункту 3.28.1.2
   30. Алгоритм переходит к пункту 3.27
   31. График – график результирующего соответствия, выводится на экран
4. **Операция разности**
   1. Создаётся пустое множество, которое ляжет в основу области отправления разности первого и второго соответствий (далее - )
   2. Создаётся пустое множество, которое ляжет в основу области прибытия разности первого и второго соответствий (далее - )
   3. Создаётся пустой график, который ляжет в основу графика разности первого и второго соответствий (далее - )
   4. Задаётся число i, равное 0
   5. Число i увеличивается на 1
   6. Задаётся число j, равное 0
   7. Число j увеличивается на 1
   8. Если i-тый элемент множества равен j-тому элементу множества , то:
      1. Если число i равно ||, алгоритм переходит к пункту 4.11
      2. Алгоритм переходит к пункту 4.5
   9. Если число j равно , то:
      1. i-тый элемент множества записывается во множество
      2. Алгоритм переходит к пункту 4.8.1
   10. Алгоритм переходит к пункту 4.7
   11. Множество – область отправления разности первого и второго соответствий, выводится на экран
   12. Число i приравнивается к 0
   13. Число i увеличивается на 1
   14. Число j приравнивается к 0
   15. Число j увеличивается на 1
   16. Если i-тый элемент множества равен j-тому элементу множества , то:
       1. Если число i равно ||, алгоритм переходит к пункту 4.19
       2. Алгоритм переходит к пункту 4.13
   17. Если число j равно , то:
       1. i-тый элемент множества записывается во множество
       2. Алгоритм переходит к пункту 4.16.1
   18. Алгоритм переходит к пункту 4.15
   19. Множество – область прибытия разности первого и второго соответствий, выводится на экран
   20. Число i приравнивается к 0
   21. Число i увеличивается на 1
   22. Число j приравнивается к 0
   23. Число j увеличивается на 1
   24. Если первый элемент i-той пары графика равен первому элементу j-той пары графика , то:
       1. Если второй элемент i-той пары графика равен второму элементу j-той пары графика , то:
          1. Если число i равно , алгоритм переходит к пункту 4.27
          2. Алгоритм переходит к пункту 4.21
   25. Если число j равно , то:
       1. i-тая пара графика записывается в график
       2. Алгоритм переходит к пункту 4.24.1.1
   26. Алгоритм переходит к пункту 4.23
   27. График – график разности первого и второго соответствий, выводится на экран
   28. Множество приравнивается к пустому множеству
   29. Множество приравнивается к пустому множеству
   30. График приравнивается к пустому графику
   31. Задаётся число i, равное 0
   32. Число i увеличивается на 1
   33. Задаётся число j, равное 0
   34. Число j увеличивается на 1
   35. Если i-тый элемент множества равен j-тому элементу множества , то:
       1. Если число i равно ||, алгоритм переходит к пункту 4.38
       2. Алгоритм переходит к пункту 4.32
   36. Если число j равно , то:
       1. i-тый элемент множества записывается во множество
       2. Алгоритм переходит к пункту 4.35.1
   37. Алгоритм переходит к пункту 4.34
   38. Множество – область отправления разности первого и второго соответствий, выводится на экран
   39. Число i приравнивается к 0
   40. Число i увеличивается на 1
   41. Число j приравнивается к 0
   42. Число j увеличивается на 1
   43. Если i-тый элемент множества равен j-тому элементу множества , то:
       1. Если число i равно ||, алгоритм переходит к пункту 4.46
       2. Алгоритм переходит к пункту 4.40
   44. Если число j равно , то:
       1. i-тый элемент множества записывается во множество
       2. Алгоритм переходит к пункту 4.43.1
   45. Алгоритм переходит к пункту 4.42
   46. Множество – область прибытия разности первого и второго соответствий, выводится на экран
   47. Число i приравнивается к 0
   48. Число i увеличивается на 1
   49. Число j приравнивается к 0
   50. Число j увеличивается на 1
   51. Если первый элемент i-той пары графика равен первому элементу j-той пары графика , то:
       1. Если второй элемент i-той пары графика равен второму элементу j-той пары графика , то:
          1. Если число i равно , алгоритм переходит к пункту 4.54
          2. Алгоритм переходит к пункту 4.48
   52. Если число j равно , то:
       1. i-тая пара графика записывается в график
       2. Алгоритм переходит к пункту 4.51.1.1
   53. Алгоритм переходит к пункту 4.50
   54. График – график разности первого и второго соответствий, выводится на экран
5. **Операция симметрической разности**
   1. Создаётся пустое множество, которое ляжет в основу области отправления результирующего соответствия (далее - )
   2. Создаётся пустое множество, которое ляжет в основу области прибытия результирующего соответствия (далее - )
   3. Создаётся пустой график, который ляжет в основу графика результирующего соответствия (далее - )
   4. Задаётся число i, равное 0
   5. Число i увеличивается на 1
   6. Задаётся число j, равное 0
   7. Число j увеличивается на 1
   8. Если i-тый элемент множества равен j-тому элементу множества , то:
      1. Если число i равно ||, алгоритм переходит к пункту 5.11
      2. Алгоритм переходит к пункту 5.5
   9. Если число j равно , то:
      1. i-тый элемент множества записывается во множество
      2. Алгоритм переходит к пункту 5.8.1
   10. Алгоритм переходит к пункту 5.7
   11. Число i приравнивается к 0
   12. Число i увеличивается на 1
   13. Число j приравнивается к 0
   14. Число j увеличивается на 1
   15. Если i-тый элемент множества равен j-тому элементу множества , то:
       1. Если число i равно ||, алгоритм переходит к пункту 5.18
       2. Алгоритм переходит к пункту 5.12
   16. Если число j равно , то:
       1. i-тый элемент множества записывается во множество
       2. Алгоритм переходит к пункту 5.15.1
   17. Алгоритм переходит к пункту 5.14
   18. Число i приравнивается к 0
   19. Число i увеличивается на 1
   20. Число j приравнивается к 0
   21. Число j увеличивается на 1
   22. Если первый элемент i-той пары графика равен первому элементу j-той пары графика , то:
       1. Если второй элемент i-той пары графика равен второму элементу j-той пары графика , то:
          1. Если число i равно , алгоритм переходит к пункту 5.25
          2. Алгоритм переходит к пункту 5.19
   23. Если число j равно , то:
       1. i-тая пара графика записывается в график
       2. Алгоритм переходит к пункту 5.22.1.1
   24. Алгоритм переходит к пункту 5.21
   25. Задаётся число i, равное 0
   26. Число i увеличивается на 1
   27. Задаётся число j, равное 0
   28. Число j увеличивается на 1
   29. Если i-тый элемент множества равен j-тому элементу множества , то:
       1. Если число i равно ||, алгоритм переходит к пункту 5.32
       2. Алгоритм переходит к пункту 5.26
   30. Если число j равно , то:
       1. i-тый элемент множества записывается во множество
       2. Алгоритм переходит к пункту 5.29.1
   31. Алгоритм переходит к пункту 5.28
   32. Множество – область отправления результирующего соответствия, выводится на экран
   33. Число i приравнивается к 0
   34. Число i увеличивается на 1
   35. Число j приравнивается к 0
   36. Число j увеличивается на 1
   37. Если i-тый элемент множества равен j-тому элементу множества , то:
       1. Если число i равно ||, алгоритм переходит к пункту 5.40
       2. Алгоритм переходит к пункту 5.34
   38. Если число j равно , то:
       1. i-тый элемент множества записывается во множество
       2. Алгоритм переходит к пункту 5.37.1
   39. Алгоритм переходит к пункту 5.36
   40. Множество – область прибытия результирующего соответствия, выводится на экран
   41. Число i приравнивается к 0
   42. Число i увеличивается на 1
   43. Число j приравнивается к 0
   44. Число j увеличивается на 1
   45. Если первый элемент i-той пары графика равен первому элементу j-той пары графика , то:
       1. Если второй элемент i-той пары графика равен второму элементу j-той пары графика , то:
          1. Если число i равно , алгоритм переходит к пункту 5.48
          2. Алгоритм переходит к пункту 5.42
   46. Если число j равно , то:
       1. i-тая пара графика записывается в график
       2. Алгоритм переходит к пункту 5.45.1.1
   47. Алгоритм переходит к пункту 5.44
   48. График – график результирующего соответствия, выводится на экран
6. **Операция инверсии**
   1. Множество – область отправления инверсии первого соответствия, выводится на экран
   2. Множество – область прибытия инверсии первого соответствия, выводится на экран
   3. Создаётся пустой график, который ляжет в основу результирующего соответствия (далее - )
   4. Задаётся число i, равное 0
   5. Число i увеличивается на 1
   6. Создаётся пустая пара p
   7. Первому элементу пары p присваивается значение второго элемента i-той пары графика
   8. Второму элементу пары p присваивается значение первого элемента i-той пары графика
   9. Пара p записывается в график
   10. Если число i меньше , алгоритм переходит к пункту 6.5
   11. График – график инверсии первого соответствия, выводится на экран
   12. Множество – область отправления инверсии второго соответствия, выводится на экран
   13. Множество – область прибытия инверсии второго соответствия, выводится на экран
   14. График приравнивается к пустому графику
   15. Число i приравнивается к 0
   16. Число i увеличивается на 1
   17. Создаётся пустая пара p
   18. Первому элементу пары p присваивается значение второго элемента i-той пары графика
   19. Второму элементу пары p присваивается значение первого элемента i-той пары графика
   20. Пара p записывается в график
   21. Если число i меньше , алгоритм переходит к пункту 6.16
   22. График – график инверсии второго соответствия, выводится на экран
7. **Операция композиции**
   1. Множество – область отправления композиции первого и второго соответствий, выводится на экран
   2. Множество – область прибытия композиции первого и второго соответствий, выводится на экран
   3. Создаётся пустой график, который ляжет в основу графика результирующего соответствия (далее – , мощность графика (далее - ) равна 0)
   4. Задаётся число i, равное 0
   5. Число i увеличивается на 1
   6. Задаётся число j, равное 0
   7. Число j увеличивается на 1
   8. Если второй элемент i-той пары графика равен первому элементу j-той пары графика , то:
      1. Создаётся пустая пара p
      2. Первому элементу пары p присваивается значение первого элемента i-той пары графика
      3. Второму элементу пары p присваивается значение второго элемента j-той пары графика
      4. Задаётся число k, равное 0
      5. Число k увеличивается на 1
      6. Если число k больше , то:
         1. увеличивается на 1
         2. Пара p записывается в график
         3. Алгоритм переходит к пункту 7.9
      7. Если первый элемент пары p равен первому элементу k-той пары графика , то:
         1. Если второй элемент пары p равен второму элементу k-той пары графика , алгоритм переходит к пункту 7.9
      8. Алгоритм переходит к пункту 7.8.5
   9. Если число j равно , то:
      1. Если число i равно ||, алгоритм переходит к пункту 7.11
      2. Алгоритм переходит к пункту 7.5
   10. Алгоритм переходит к пункту 7.7
   11. График – график композиции первого и второго соответствий, выводится на экран
   12. Множество – область отправления композиции второго и первого соответствий, выводится на экран
   13. Множество – область прибытия композиции второго и первого соответствий, выводится на экран
   14. График приравнивается к пустому графику ()
   15. Число i приравнивается к 0
   16. Число i увеличивается на 1
   17. Задаётся число j, равное 0
   18. Число j увеличивается на 1
   19. Если второй элемент i-той пары графика равен первому элементу j-той пары графика , то:
       1. Создаётся пустая пара p
       2. Первому элементу пары p присваивается значение первого элемента i-той пары графика
       3. Второму элементу пары p присваивается значение второго элемента j-той пары графика
       4. Задаётся число k, равное 0
       5. Число k увеличивается на 1
       6. Если число k больше , то:
          1. увеличивается на 1
          2. Пара p записывается в график
          3. Алгоритм переходит к пункту 7.20
       7. Если первый элемент пары p равен первому элементу k-той пары графика , то:
          1. Если второй элемент пары p равен второму элементу k-той пары графика , алгоритм переходит к пункту 7.20
       8. Алгоритм переходит к пункту 7.19.5
   20. Если число j равно , то:
       1. Если число i равно ||, алгоритм переходит к пункту 7.22
       2. Алгоритм переходит к пункту 7.16
   21. Алгоритм переходит к пункту 7.18
   22. График – график композиции второго и первого соответствий, выводится на экран
8. **Операция сужения**
   1. Создаётся пустое множество, которое ляжет в основу множества, на которое будет выполняться сужение первого соответствия (далее – множество )
   2. Задаётся множество
   3. Задаётся множество
   4. Задаётся график
   5. Пользователь вводит число, соответствующее тому соответствию, сужение которого нужно выполнить – 1 или 2
      1. Если пользователь ввёл число 1, то:
         1. Множество приравнивается к
         2. Множество приравнивается к
         3. График приравнивается к
      2. Если пользователь ввёл число 2, то:
         1. Множество приравнивается к
         2. Множество приравнивается к
         3. График приравнивается к
   6. Пользователь вводит мощность множества (далее - )
   7. Если больше , то:
      1. На экран выводится сообщение о том, что мощность множества, на которое выполняется сужение, не может превышать мощности области отправления
      2. Алгоритм переходит к пункту 8.6
   8. Задаётся число i, равное 0
   9. Число i увеличивается на 1
   10. Пользователь вводит i-тый элемент множества
   11. Задаётся число j, равное 0
   12. Число j увеличивается на 1
   13. Если число j равно , то:
       1. На экран выводится сообщение о том, что введённого пользователем элемента нет во множестве
       2. Алгоритм переходит к пункту 8.1
   14. Если i-тый элемент множества не равен j-тому элементу множества , алгоритм переходит к пункту 8.12
   15. Если число i равно , алгоритм переходит к пункту 8.17
   16. Алгоритм переходит к пункту 8.9
   17. Задаётся пустой график, который ляжет в основу графика результирующего соответствия (далее - )
   18. Число i приравнивается к 0
   19. Число i увеличивается на 1
   20. Число j приравнивается к 0
   21. Число j увеличивается на 1
   22. Если первый элемент j-той пары графика равен i-тому элементу множества , j-тая пара графика записывается в график
   23. Если число j равно , то:
       1. Если число i равно , алгоритм переходит к пункту 8.25
       2. Алгоритм переходит к пункту 8.19
   24. Алгоритм переходит к пункту 8.21
   25. Множество – область отправления сужения выбранного пользователем соответствия на множество , выводится на экран
   26. Множество – область прибытия сужения выбранного пользователем соответствия на множество , выводится на экран
   27. График – график сужения выбранного пользователем соответствия на множество , выводится на экран
9. **Операция продолжения**
   1. Задаётся множество
   2. Задаётся множество
   3. Задаётся график
   4. Пользователь вводит число, соответствующее тому соответствию, продолжение которого нужно выполнить – 1 или 2
      1. Если пользователь ввёл число 1, то:
         1. Множество приравнивается к
         2. Множество приравнивается к
         3. График приравнивается к
      2. Если пользователь ввёл число 2, то:
         1. Множество приравнивается к
         2. Множество приравнивается к
         3. График приравнивается к
   5. Создаётся пустой график, который ляжет в основу графика результирующего соответствия (далее - )
   6. Пользователь вводит мощность графика (далее - )
   7. Если меньше , то:
      1. На экран выводится сообщение о том, что мощность графика продолжения соответствия не может быть меньше мощности графика исходного соответствия
      2. Алгоритм переходит к пункту 9.6
   8. Задаётся число g, равное 0
   9. Задаётся число i, равное 0
   10. Число i увеличивается на 1
   11. Создаётся пустая пара p
   12. Пользователь вводит первый и второй элементы пары p
   13. Задаётся число j, равное 0
   14. Число j увеличивается на 1
   15. Если число j равно , то:
       1. На экран выводится сообщение о том, что первого элемента введённой пользователем пары нет в области отправления соответствия
       2. Алгоритм переходит к пункту 9.11
   16. Если первый элемент пары p не равен j-тому элементу множества алгоритм переходит к пункту 9.14
   17. Число j приравнивается к 0
   18. Число j увеличивается на 1
   19. Если число j равно то:
       1. На экран выводится сообщение о том, что второго элемента введённой пользователем пары нет в области прибытия соответствия
       2. Алгоритм переходит к пункту 9.11
   20. Если второй элемент пары p не равен j-тому элементу множества , алгоритм переходит к пункту 9.18
   21. Число j приравнивается к 0
   22. Число j увеличивается на 1
   23. Если число j равно g, то:
       1. Число g увеличивается на 1
       2. Пара p записывается в график
       3. Алгоритм переходит к пункту 9.26
   24. Если первый элемент пары p равен первому элементу j-той пары графика , то:
       1. Если второй элемент пары p равен второму элементу j-той пары графика , то:
          1. На экран выводится сообщение о том, что введённая пользователем пара уже есть в графике G
          2. Алгоритм переходит к пункту 9.11
   25. Алгоритм переходит к пункту 9.22
   26. Если число i меньше , алгоритм переходит к пункту 9.10
   27. Число i приравнивается к 0
   28. Число i увеличивается на 1
   29. Число j приравнивается к 0
   30. Число j увеличивается на 1
   31. Если первый элемент i-той пары графика равен первому элементу j-той пары графика , то:
       1. Если второй элемент i-той пары графика равен второму элементу j-той пары графика , то:
          1. Если число i равно , алгоритм переходит к пункту 9.34
          2. Алгоритм переходит к пункту 9.28
   32. Если число j равно , то:
       1. На экран выводится сообщение о том, что график не является подмножеством введённого пользователем графика
       2. Алгоритм переходит к пункту 9.6
   33. Алгоритм переходит к пункту 9.30
   34. Множество – область отправления продолжения выбранного пользователем соответствия, выводится на экран
   35. Множество – область прибытия продолжения выбранного пользователем соответствия, выводится на экран
   36. График – график продолжения выбранного пользователем соответствия, выводится на экран
   37. Алгоритм завершает работу