### Постановка задачи:

Даны два соответствия. Найти их объединение, пересечение, разность, симметричную разность, инверсию, композицию, образ, прообраз, сужение и продолжение.   
Графики, области отправления и области прибытия соответствий задаются методом перечисления.

### Уточнение постановки задачи

За один проход можно выполнить одну операцию выбранную пользователем.

Элементы обоих соответсвий вводятся пользователем с клавиатуры.

Используются два соответствия: Г1=<X, Y, G> и Г2=<U, V, F>.

Мощность множеств X, Y, U, V задаётся пользователем в промежутке [1;10].

Мощность графиков G и F задаётся пользователем в промежутке [1;10].

Элементами множеств X, Y, U, V являются целые числа в промежутке [1;30].

Элементами графика G являются кортежи, первая компонента которых принадлежит множеству X, а вторая – множеству Y, обе компоненты кортежей являются натуральными числами в промежутке [1;30].

Элементами графика F являются кортежи, первая компонента которых принадлежит множеству U, а вторая – множеству V, обе компоненты кортежей являются натуральными числами в промежутке [1;30].

Операция инверсии выполняется для соответсвия Г1.

Операция композиции и разности выполняется для соответствий Г1 и Г2.

Операции сужения, продолжения, образа и прообраза выполняются только для соответствия Г1.

Для операции сужения, образа и прообраза пользователь сам задаёт множество А перечислением.

Мощность множества А задается пользователем в промежутке [1;10].

Элементами множества А являются натуральные числа в промежутке [1;30].

Графиком расширения является декартовое произведение области отправления и области прибытия соответствия Г1.

### Определения:

*Множество* – любое собрание определенных и различных между собой объектов нашей интуиции или интеллекта, мыслимое как единое целое.

*Мощность* множества А обозначается как |А| - это число элементов множества.

*Кортеж* - упорядоченный n-набор <a1, ..., аn> на множествах А1, ..., Аn, характеризующийся не только входящими в него элементами a1А1, ..., аnАn, но и порядком, в котором они перечисляются.

*График* — это множество пар, т.е. множество, каждый элемент которого является парой или кортежем длины 2. Множество Р называется графиком, если каждый элемент его пара.

*Объединением* соответствий A = <X, Y, G> и B = <U, V, F> называют соответствие AB = <XU, YV, GF>.

*Пересечением* соответствий A = <X, Y, G> и B = <U, V, F> называют соответствие AB = <XU, YV, GF>.

*Разностью* соответствий A = <X, Y, G> и B = <U, V, F> называют соответствие A\B = <X\U, Y\V, G\F>

*Инверсией* соответствия A= <X, Y, G> является соответствие A-1, такое, что множество Y является областью отправления соответствия A-1; множество X является областью прибытия соответствия A-1, а график соответствия G-1 является инверсией графика G соответствия A.

*Композицией* (произведением) соответствий A = <X, Y, G> и B= <U, V, F>называют соответствие A·B= <X, V, G·F>.

*Образом* множества M при соответствии A = (Х, Y, F) называется множество A(M) = {у|(х,у)ϵ F и хϵ M}.

*Прообразом* множества N при соответствии A = (Х, Y, F) называется множество A-1 (N) = {х|(х,у)ϵ F и уϵ N}.

*Сужением* соответствия A на множестве W называется соответствие AW=<X, Y, F(W×Y)>.

*Продолжением* соответствия A= (Х, Y, F) называется соответствие K=(Z,U,H), причем F H, Z=X, U=Y.

### 4) Описание алгоритма:

##### Ввод данных:

## Пользователь задаёт мощность областиотправления Х соответствия Г1.

## 1.2) Пользователь вводит элементы области отправления Х.

## 1.3) Пользователь задаёт мощность области прибытия У соответствия Г1.

## 1.4) Пользователь вводит элементы области прибытия У.

## 1.5) Пользователь задаёт мощность графика G соответствия Г1.

1.6) Пользователь вводит элементы графика G.

1.7) Cоответствие Г1 заполнено элементами и выводится на экран.

## 1.8) Пользователь задаёт мощность областиотправления U соответствия Г2.

## 1.9) Пользователь вводит элементы области отправления U.

## 1.10) Пользователь задаёт мощность области прибытия V соответствия Г2.

## 1.11) Пользователь вводит элементы области прибытия V.

## 1.12) Пользователь задаёт мощность графика F соответствия Г2.

1.13) Пользователь вводит элементы графика F.

1.14) Cоответствие Г2 заполнено элементами и выводится на экран.

##### 2) Выбор операции:

2.1) Пользователь должен выбрать, какую из операции он хочет выполнить, в зависимости от его выбора будет выполнена операция из следующего списка:

* Объединение
* Пересечение
* Разность соответствий Г1 и Г2
* Симметрическая разность
* Инверсия соответсвия Г1
* Композиция соответствий Г1 и Г2
* Образ соответствия Г1
* Прообраз соответсвия Г1
* Сужение соответсвия Г1
* Продолжение соответсвия Г1

Если пользователь выбрал объединение, переходим к пункту 3;

Если пользователь выбрал пересечение, переходим к пункту 4;

Если пользователь выбрал разность соответствий Г1 и Г2, переходим к пункту 5;

Если пользователь выбрал симметрическую разность, переходим к пункту 6;

Если пользователь выбрал инверсию соответсвия Г1, переходим к пункту 7;

Если пользователь выбрал композицию соответствий Г1 и Г2, переходим к пункту 8;

Если пользователь выбрал образ соответствия Г1, переходим к пункту 9;

Если пользователь выбрал прообраз соответсвия Г1, переходим к пункту 10;

Если пользователь выбрал сужение соответсвия Г1, переходим к пункту 11;

Если пользователь выбрал продолжение соответсвия Г1, переходим к пункту 12;

##### 3) Операция объединения

3.1) Создается пустое множество Сx, которое будет результатом объединения Х и U

3.2) Записываем все элементы множества X в множество Сx

3.3) Выбираем первый элемент множества U.

3.4) Выбираем первый элемент множества X

3.5) Если выбранный элемент множества X не равен выбранному элементу множества U то:

3.5.1) Если выбранный элемент множества X является последним, то записываем выбранный элемент множества U во множество Cx:

3.5.1.1) Переходим к пункту (3.6.1)

3.5.2) Выбираем следующий элемент множества X:

3.5.2.1) Переходим к пункту (3.5)

3.6) Если рассматриваемый элемент множества X равен рассматриваемому элементу множества U, то:

3.6.1) Если элемент U является последним

3.6.1.1) Переходим к пункту (3.7)

3.6.2) Рассмотрим следующий элемент множества U

3.6.2.1) Переходим к пункту (3.4)

3.7) Множество Сx является множеством объединения множеств X и U, выводится на экран

3.8) Создается пустое множество Сy, которое будет результатом объединения Y и V

3.9) Записываем все элементы множества Y в множество Сy

3.10) Выбираем первый элемент множества V.

3.11) Выбираем первый элемент множества Y

3.12) Если выбранный элемент множества Y не равен выбранному элементу множества V то:

3.12.1) Если выбранный элемент множества Y является последним, то записываем выбранный элемент множества V во множество Cy:

3.12.1.1) Переходим к пункту (3.13.1)

3.12.2) Выбираем следующий элемент множества Y:

3.12.2.1) Переходим к пункту (3.12)

3.13) Если рассматриваемый элемент множества Y равен рассматриваемому элементу множества V, то:

3.13.1) Если элемент V является последним

3.13.1.1) Переходим к пункту (3.14)

3.13.2) Рассмотрим следующий элемент множества V

3.13.2.1) Переходим к пункту (3.11)

3.14) Множество Сy является множеством объединения множеств Y и V, выводится на экран

3.15) Создается пустой график С, который будет результатом операции объединения графиков G и F

3.16) Записываем все элементы графика G в график С.

3.17) Выбираем первый элемент графика F.

3.18) Выбираем первый элемент графика G.

3.19) Если первая компонента элемента графика G равна первой компоненте элемента графика F.

3.19.1) Если вторая компонента элемента графика G равна второй компоненте элемента графика F.

3.19.1.1) Если элемент F является последним.

3.19.1.1.1) Переходим к пункту (3.22)

3.19.1.2) Рассмотрим следующий элемент графика F.

3.19.1.2.1) Переходим к пункту (3.18)

3.20) Если выбранный элемент графика G является последним, то записываем выбранный элемент графика F в график С:

3.20.1) Переходим к пункту (3.19.1.1)

3.21) Выбираем следующий элемент графика G:

3.21.1) Переходим к пункту (3.19)

3.22) График С является объединением графиков G и F, выводится на экран.

3.23) Программа завершает работу.

##### 4) Операция пересечения

4.1) Создается пустое множество Dx, которое будет результатом операции пересечения множеств X и U

4.2) Выбираем первый элемент множества X.

4.3) Выбираем первый элемент множества U.

4.4) Если выбранный элемент множества X равен выбранному элементу множества U, то элемент множества U записываем в множества Dx.

4.5) Если выбранный элемент множества U является последним:

4.5.1) Переходим к пункту (4.7)

4.6) Выбираем следующий элемент множества U:

4.6.1) Переходим к пункту (4.4)

4.7) Если выбранный элемент множества X является последним:

4.7.1) Переходим к пункту (4.9)

4.8) Выбираем следующий элемент множества X

4.8.1) Преходим к пункту (4.4)

4.9) Dx – результат пересечения множеств X и U, Dx выводится на экран

4.10) Создается пустое множество Dy, которое будет результатом операции пересечения множеств Y и V

4.11) Выбираем первый элемент множества Y.

4.12) Выбираем первый элемент множества V.

4.13) Если выбранный элемент множества Y равен выбранному элементу множества V, то элемент множества V записываем в множества D.

4.14) Если выбранный элемент множества V является последним:

4.14.1) Переходим к пункту (4.16)

4.15) Выбираем следующий элемент множества V:

4.15.1) Переходим к пункту (4.13)

4.16) Если выбранный элемент множества Y является последним:

4.16.1) Переходим к пункту (4.18)

4.17) Выбираем следующий элемент множества Y

4.17.1) Преходим к пункту (4.13)

4.18) Dy – результат пересечения множеств Y и V, Dy выводится на экран

4.19) Создается пустой график D, который будет результатом операции перечения графиков G и F

4.20) Выбираем первый элемент графика G.

4.21) Выбираем первый элемент графика F.

4.22) Если первая компонента элемента графика F равна первой компоненте элемента графика G.

4.22.1) Если вторая компонента элемента графика F равна второй компоненте элемента графика G.

4.22.1.1) Элемент графика F записываем в график D.

4.22.2) Если выбранный элемент множества F является последним:

4.22.2.1) Переходим к пункту (4.24)

4.23) Выбираем следующий элемент графика F.

4.23.1) Переходим к пункту (4.22)

4.24) Если выбранный элемент множества G является последним:

4.24.1) Переходим к пукнту (4.26)

4.25) Выбираем следующий элемент графика G

4.25.1) Переходим к пункту (4.22)

4.26) D – результат пересечения множеств G и F, D выводится на экран

4.27) Программа завершает работу

##### 5) Операция разности Г1\Г2

5.1) Создается пустое множество Ex, которое будет результатом операции разности множеств X и U

5.2) Выбираем первый элемент множества X

5.3) Выбираем первый элемент множества U

5.4) Если выбранный элемент множества X не равен выбранному элементу множества U:

5.4.1) Если выбранный элемент множества U не является последним:

5.4.1.1) Если выбранный элемент множества X не является последним, выбираем следующий элемент множества U

5.4.1.1.1) Переходим к пункту (5.4)

5.4.1.2) Если выбранный элемент множества X является последним

5.4.1.2.1) Переходим к пункту (5.6)

5.4.2) Если выбранный элемент множества U является последним, записываем выбранный элемент множества X в множество Еx

5.4.3) Выбираем следующий элемент множества X

5.4.3.1) Переходим к пункту (5.3)

5.5) Если выбранный элемент множества X равен выбранному элементу множества U

5.5.1) Если элемент не является последним

5.5.2) Выбираем следующий элемент множества X

5.5.3) Переходим к пункту (5.3)

5.6) Еx – результат разности множества X и U, Еx выводится на экран

5.7) Создается пустое множество Ey, которое будет результатом операции разности множеств Y и V

5.8) Выбираем первый элемент множества Y

5.9) Выбираем первый элемент множества V

5.10) Если выбранный элемент множества Y не равен выбранному элементу множества V:

5.10.1) Если выбранный элемент множества V не является последним:

5.10.1.1) Если выбранный элемент множества Y не является последним, выбираем следующий элемент множества V

5.10.1.1.1) Переходим к пункту (5.10)

5.10.1.2) Если выбранный элемент множества Y является последним

5.10.1.2.1) Переходим к пункту (5.12)

5.10.2) Если выбранный элемент множества V является последним, записываем выбранный элемент множества Y в множество Еy

5.10.3) Выбираем следующий элемент множества Y

5.10.3.1) Переходим к пункту (5.9)

5.11) Если выбранный элемент множества Y равен выбранному элементу множества V

5.11.1) Если элемент не является последним

5.11.2) Выбираем следующий элемент множества Y

5.11.3) Переходим к пункту (5.9)

5.12) Еy – результат разности множества Y и V, Еy выводится на экран

5.13) Создается пустое график E, который будет результатом операции разности графиков G и F

5.14) Выбираем первый элемент графика G

5.15) Выбираем первый элемент графика F

5.16) Если первая компонента элемента графика F равна первой компоненте элемента графика G.

5.16.1) Если вторая компонента элемента графика F равна второй компоненте элемента графика G.

5.16.1.1) Если элемент не является последним

5.16.1.1.1) Выбираем следующий элемент графика G

5.16.1.1.2) Переходис к пукнту (5.15)

5.16.1.2) Переходим к пункту (5.19)

5.17) Если выбранный элемент графика F не является последним:

5.17.1) Если выбранный элемент графика G не является последним, выбираем следующий элемент графика F

5.17.1.1) Переходим к пункту (5.16)

5.17.2) Если выбранный элемент графика G является последним

5.17.2.1) Переходим к пункту (5.19)

5.18) Если выбранный элемент графика F является последним, записываем выбранный элемент графика G в график Е

5.18.1) Выбираем следующий элемент графика G

5.18.1.1) Переходим к пункту (5.15)

5.19) Е – результат разности графиков G и F, Е выводится на экран

5.20) Программа завершает работу

##### Операция симметрической разности

6.1) Создается пустое множество Hx, которое будет результатом операции симметрической разности множеств X и U

6.2) Выбираем первый элемент множества X

6.3) Выбираем первый элемент множества U

6.4) Если выбранный элемент множества X не равен выбранному элементу множества U:

6.4.1) Если выбранный элемент множества U не является последним:

6.4.1.1) Если выбранный элемент множества X не является последним, выбираем следующий элемент множества U

6.4.1.1.1) Переходим к пункту (6.4)

6.4.1.2) Если выбранный элемент множества X является последним

6.4.1.2.1) Переходим к пункту (6.6)

6.4.2) Если выбранный элемент множества U является последним, записываем выбранный элемент множества X в множество Hx

6.4.3) Выбираем следующий элемент множества X

6.4.3.1) Переходим к пункту (6.3)

6.5) Если выбранный элемент множества X равен выбранному элементу множества U

6.5.1) Если элемент не является последним

6.5.1) Выбираем следующий элемент множества X

6.5.2) Переходим к пункту (6.3)

6.6) Выбираем первый элемент множества U

6.7) Выбираем первый элемент множества X

6.8) Если выбранный элемент множества U не равен выбранному элементу множества X:

6.8.1) Если выбранный элемент множества X не является последним:

6.8.1.1) Если выбранный элемент множества U не является последним, выбираем следующий элемент множества X

6.8.1.1.1) Переходим к пункту (6.8)

6.8.1.2) Если выбранный элемент множества U является последним

6.8.1.2.1) Переходим к пункту (6.10)

6.8.2) Если выбранный элемент множества X является последним, записываем выбранный элемент множества U в множество Hx

6.8.3) Выбираем следующий элемент множества U

6.8.3.1) Переходим к пункту (6.7)

6.9) Если выбранный элемент множества U равен выбранному элементу множества X

6.9.1) Если элемент не является последним

6.9.2) Выбираем следующий элемент множества U

6.9.3) Переходим к пункту (6.7)

6.10) Hx – результат симметрической разности множества X и U, Hx выводится на экран

6.11) Создается пустое множество Hy, которое будет результатом операции симметрической разности множеств Y и V

6.12) Выбираем первый элемент множества Y

6.13) Выбираем первый элемент множества V

6.14) Если выбранный элемент множества Y не равен выбранному элементу множества V:

6.14.1) Если выбранный элемент множества V не является последним:

6.14.1.1) Если выбранный элемент множества Y не является последним, выбираем следующий элемент множества V

6.14.1.1.1) Переходим к пункту (6.14)

6.14.1.2) Если выбранный элемент множества Y является последним

6.14.1.2.1) Переходим к пункту (6.16)

6.14.2) Если выбранный элемент множества V является последним, записываем выбранный элемент множества Y в множество Hy

6.14.3) Выбираем следующий элемент множества Y

6.14.3.1) Переходим к пункту (6.13)

6.15) Если выбранный элемент множества Y равен выбранному элементу множества V

6.15.1) Если элемент не является последним

6.15.1) Выбираем следующий элемент множества Y

6.15.2) Переходим к пункту (6.13)

6.16) Выбираем первый элемент множества V

6.17) Выбираем первый элемент множества Y

6.18) Если выбранный элемент множества V не равен выбранному элементу множества Y:

6.18.1) Если выбранный элемент множества Y не является последним:

6.18.1.1) Если выбранный элемент множества V не является последним, выбираем следующий элемент множества Y

6.18.1.1.1) Переходим к пункту (6.18)

6.18.1.2) Если выбранный элемент множества V является последним

6.18.1.2.1) Переходим к пункту (6.20)

6.18.2) Если выбранный элемент множества Y является последним, записываем выбранный элемент множества V в множество Hy

6.18.3) Выбираем следующий элемент множества V

6.18.3.1) Переходим к пункту (6.17)

6.19) Если выбранный элемент множества V равен выбранному элементу множества Y

6.19.1) Если элемент не является последним

6.19.2) Выбираем следующий элемент множества V

6.19.3) Переходим к пункту (6.17)

6.20) Hy – результат симметрической разности множества Y и V, Hy выводится на экран

6.21) Создается пустой график H, который будет результатом операции симметрической разности графиков G и F

6.22) Выбираем первый элемент графика G

6.23) Выбираем первый элемент графика F

6.24) Если первая компонента элемента графика F равна первой компоненте элемента графика G.

6.24.1) Если вторая компонента элемента графика F равна второй компоненте элемента графика G.

6.24.1.1) Если элемент не является последним

6.24.1.1.1) Выбираем следующий элемент графика G

6.24.1.1.2) Переходис к пукнту (6.23)

6.24.1.2) Переходим к пункту (6.27)

6.25) Если выбранный элемент графика F не является последним:

6.25.1) Если выбранный элемент графика G не является последним, выбираем следующий элемент графика F

6.25.2.1) Переходим к пункту (6.24)

6.25.2) Если выбранный элемент графика G является последним

6.25.2.1) Переходим к пункту (6.27)

6.26) Если выбранный элемент графика F является последним, записываем выбранный элемент графика G в график H

6.26.1) Выбираем следующий элемент графика G

6.26.1.1) Переходим к пункту (6.23)

6.27) Выбираем первый элемент графика F

6.28) Выбираем первый элемент графика G

6.29) Если первая компонента элемента графика G равна первой компоненте элемента графика F.

6.29.1) Если вторая компонента элемента графика G равна второй компоненте элемента графика F.

6.29.1.1) Если элемент не является последним

6.29.1.1.1) Выбираем следующий элемент графика F

6.29.1.1.2) Переходис к пукнту (6.28)

6.29.1.2) Переходим к пункту (6.32)

6.30) Если выбранный элемент графика G не является последним:

6.30.1) Если выбранный элемент графика F не является последним, выбираем следующий элемент графика G

6.30.1.1) Переходим к пункту (6.29)

6.30.2) Если выбранный элемент графика F является последним

6.30.2.1) Переходим к пункту (6.32)

6.31) Если выбранный элемент графика G является последним, записываем выбранный элемент графика F в график H

6.31.1) Выбираем следующий элемент графика F

6.31.1.1) Переходим к пункту (6.28)

6.32) H – результат симметрической разности графиков G и F, H выводится на экран

6.33) Программа завершает работу

##### Операция инверсии соответствия Г1

7.1) Создаём пустое множество Ix, которое будет является областью отправления соответствия Г1.

7.2) Выбираем первый элемент Y.

7.3) Записываем выбранный элемент Y в множество Ix.

7.4) Если выбранный элемент Y последний

7.4.1) Переходим к пункту (7.7)

7.5) Выбираем следующий элемент Y.

7.6) Переходим к пункту (7.3)

7.7) Ix - область отправления нового соответствия Г1-1, Nх выводится на экран.

7.8) Создаём пустое множество Iy, которое будет является областью прибытия соответствия Г1.

7.9) Выбираем первый элемент X.

7.10) Записываем выбранный элемент X в множество Iy.

7.11) Если выбранный элемент X последний

7.11.1) Переходим к пункту (7.14)

7.12) Выбираем следующий элемент X.

7.13) Переходим к пункту (7.10)

7.14) Iy - область отправления нового соответствия Г1-1, Iy выводится на экран.

7.15) Создается пустой график I, который будет результатом операции инверсии графика G

7.16) Выбираем первый элемент графика G

7.17) Создаем пару i

7.18) Записываем первую компоненту элемента графика G на место второй компоненты i

7.19) Записываем вторую компоненту элемента графика G на место первой компоненты i

7.20) Добавляем пару i в график I

7.21) Если выбранный элемент графика G был последним

7.21.1) Переходим к пунту (7.23)

7.22) Выбираем следующий элемент графика G

7.22.1) Переходим к пункту (7.17)

7.23) I – график нового соответствия Г1-1, I выводится на экран

7.24) Программа завершает работу

##### Операция композиции Г1\*Г2

8.1) Создаём пустое множество Jх, которое является областью отправления нового соответствия.

8.2) Выбираем первый элемент X.

8.3) Записываем выбранный элемент X в множество Jx.

8.4) Если выбранный элемент X последний

8.4.1) Переходим к пункту (8.7)

8.5) Выбираем следующий элемент X.

8.6) Переходим к пункту (8.3)

8.7) Jx - область отправления нового соответствия Г1· Г2, Jх выводится на экран.

8.8) Создаём пустое множество Jy, которое является областью прибытия нового соответствия.

8.9) Выбираем первый элемент V

8.10) Записываем выбранный элемент V в множество Jy

8.11) Если выбранный элемент V последний

8.11.1) Переходим к пункту (8.14)

8.12) Выбираем следующий элемент V.

8.13) Переходим к пункту (8.10)

8.14) Jy - область отправления нового соответствия Г1· Г2, Jy выводится на экран.

8.15) Создается пустой график J, который будет результатом операции композиции графиков G и F

8.16) Выбираем первый элемент графика G

8.17) Выбираем первый элемент графика F

8.18) Если вторая компонента графика G равна первой компоненте графика F

8.18.1) Создаем пару j

8.18.2) Записываем первую компоненту графика G на место первой компоненты j

8.18.3) Записываем вторую компоненту графика F на место второй компоненты j

8.18.3.1) Если график J пустой

8.18.3.1.1) Переходим к пункту (8.18.4)

8.18.3.2) Выбираем первый элемент графика J

8.18.3.3) Если первая компонента пары j равна первой компоненте выбранного элемента из графика J

8.18.3.3.1) Если вторая компонента пары j равна второй компоненте выбранного элемента из графика J

8.18.3.3.1.1) Переходим к пункту (8.18.5)

8.18.3.4) Если выбранный элемент графика J является последним

8.18.3.4.1) Переходим к пункту (8.18.4)

8.18.3.5) Выбираем следующий элемент графика J

8.18.3.5.1) Переходим к пункту (8.18.3.3)

8.18.4) Записываем пару j в график J

8.18.5) Если выбранный элемент графика F является последним

8.18.5.1) Переходим к пункту (8.18.7)

8.18.6) Выбираем следующий элемент графика F

8.18.6.1) Переходим к пункту (8.18)

8.18.7) Если выбранный элемент графика G был последним

8.18.7.1) Переходим к пункту (8.20)

8.18.8) Выбираем следующий элемент графика G

8.18.8.1) Переходим к пукнту (8.17)

8.19) Если вторая компонента графика G не равна первой компоненте графика F

8.19.1) Переходим к пункту (8.18.5)

8.20) J – график нового соответствия Г1· Г2, J выводится на экран

8.21) Программа завершает работу

##### Образ соответствия Г1

9.1) Создаём пустое множество A.

9.2) Пользователь вводит мощность множества A.

9.3) Пользователь вводит с клавиатуры элементы A.

9.4) Создаём пустое множество B, который является результатом операции.

9.5) Выбираем первый элемент первого кортежа графика X.

9.6) Выбираем первый элемент множества A.

9.7) Сравниваем выбранные элементы.

9.8) Если выбранные элементы равны, то:

9.8.1) Если множество B пустое

9.8.1.1) Переходим к пункту (9.8.8)

9.8.2) Выбираем первый элемент множества B.

9.8.3) Сравниваем выбранный элемент множества B и второй элемент выбранного  
кортежа графика X.

9.8.4) Если выбранные элементы равны

9.8.4.1) Переходим к пункту (9.9)

9.8.5) Если выбранный элемент A последний

9.8.5.1) Переходим к пункту (9.8.8)

9.8.6) Выбираем следующий элемент B.

9.8.7) Переходим к пункту (9.8.3)

9.8.8) Записываем второй элемент выбранного кортежа X в множество B.

9.9) Если выбранный кортеж графика X последний

9.9.1) Переходим к пункту (9.12)

9.10) Выбираем первый элемент следующего кортежа X.

9.11) Переходим к пункту (9.7)

9.12) Если выбранный элемент множества A является последним

9.12.1) Переходим к пункту (9.16)  
  
9.13) Выбираем первый элемент следующего кортежа X.

9.14) Выбираем следующий элемент A.

9.15) Переходим к пункту (9.7)

9.16) B–образ соответствия Г1, В выводится на экран.

9.17) Программа завершает работу

##### Прообраз соответствия Г1

10.1) Создаём пустое множество A.

10.2) Пользователь вводит мощность множества A.

10.3) Пользователь вводит с клавиатуры элементы A.

10.4) Создаём пустое множество K, который является результатом операции.

10.5) Выбираем второй элемент первого кортежа графика X.

10.6) Выбираем первый элемент множества A.

10.7) Сравниваем выбранные элементы.

10.8) Если выбранные элементы равны, то:

10.8.1) Если множество B пустое

10.8.1.1) Переходим к пункту (10.8.8)

10.8.2) Выбираем первый элемент множества K.

10.8.3) Сравниваем выбранный элемент множества K и первый элемент выбранного  
кортежа графика X.

10.8.4) Если выбранные элементы равны

10.8.4.1) Переходим к пункту (10.9)

10.8.5) Если выбранный элемент A последний

10.8.5.1) Переходим к пункту (10.8.8)

10.8.6) Выбираем следующий элемент B.

10.8.7) Переходим к пункту (10.8.3)

10.8.8) Записываем первый элемент выбранного кортежа X в множество K.

10.9) Если выбранный кортеж графика X последний

10.9.1) Переходим к пункту (10.12)

10.10) Выбираем второй элемент следующего кортежа X.

10.11) Переходим к пункту (10.7)

10.12) Если выбранный элемент множества A является последним

10.12.1) Переходим к пункту (10.16)  
  
10.13) Выбираем второй элемент следующего кортежа X.

10.14) Выбираем следующий элемент A.

10.15) Переходим к пункту (10.7)

10.16) K–прообраз соответствия Г1, K выводится на экран.

10.17) Программа завершает работу

##### Сужение соответствия Г1

11.1) Создаём пустой график L, который будет результатом операции.

11.2) Создаём пустое множество A.

11.3) Пользователь вводит мощность множества A.

11.4) Пользователь вводит с клавиатуры элементы A.

11.5) Выбираем первый элемент первого кортежа графика G.

11.6) Выбираем первый элемент множества A.

11.7) Сравниваем выбранные элементы.

11.8) Если выбранные элементы равны, то записываем выбранный кортеж G в график L.

11.9) Если выбранный кортеж графика G последний

11.9.1) Переходим к пункту (11.12)

11.10) Выбираем первый элемент следующего кортежа G.

11.11) Переходим к пункту (11.7)

11.12) Если выбранный элемент множества A является последним

11.12.1) Переходим к пункту (11.16)

11.13) Выбираем первый элемент следующего кортежа G.

11.14) Выбираем следующий элемент A.

11.15) Переходим к пункту (11.7)

11.16) L–график сужение соответствия Г1; Х, Y, L выводится на экран.

11.17) Программа завершает работу

##### Продолжение соответствия Г1

12.1) Создаём пустой график M, который является результатом операции.

12.2) Выбираем первый элемент множества Х.

12.3) Выбираем первый элемент множества Y.

12.4) Записываем выбранный элемент множества X в первую компоненту кортежа длины

12.5) Выбранный элемент множества Y во вторую компонетну того же кортежа.

12.6) Полученый кортеж записываем в множесво M.

12.7) Если выбранный элемент множества Y был последним

12.7.1) Переходим к пункту (12.10)

12.8) Выбираем следующий элемент множесва Y.

12.9) Переходим к пункту (12.4)

12.10) Если выбранный элемен множесва X был последним

12.10.1) Переходим к пункту (12.13)

12.11) Выбираем следующий элемент множества X.

12.12) Переходим к пункту (12.3)

12.13) RG–график расширения соответствия Г1; Х, Y, M выводится на экран

12.14) Программа завершает работу