Белорусский государственный университет информатики и радиоэлектроники

**Лабораторная работа №1**

«Множества. Операции над множествами»

Подготовил:

Захаренков И. Д.

Гр.421702

**Постановка задачи:**

**Цель**: Разработать алгоритм и реализовать программу для выполнения базовых операций над множествами.

**Задачи**:

1. Разработать алгоритм для ввода двух множеств.
2. Реализовать операции объединения, пересечения, разности, симметрической разности и дополнения множеств.
3. Вывести результаты всех операций.

**Уточнение постановки задачи:**

1. Элементами множеств являются натуральные числа от 1 до 100.
2. Мощность каждого множества задается пользователем и не превышает 10 элементов.
3. Элементы множеств вводятся пользователем.
4. Универсальное множество U содержит все натуральные числа от 1 до 100.
5. Множества могут быть заданы перечислением или с помощью формулы (в данной реализации используется только ввод пользователя).

**Определения:**

* Множество - совокупность различных объектов, рассматриваемых как единое целое.
* Мощность множества - количество элементов в множестве.
* Объединение множеств A и B (A ∪ B) - множество, содержащее все элементы, принадлежащие хотя бы одному из множеств A или B.
* Пересечение множеств A и B (A ∩ B) - множество, содержащее элементы, принадлежащие одновременно обоим множествам A и B.
* Разность множеств A и B (A \ B) - множество, содержащее элементы множества A, не принадлежащие множеству B.
* Симметрическая разность множеств A и B (A Δ B) - множество элементов, принадлежащих либо A, либо B, но не обоим множествам одновременно.
* Дополнение множества A (A') - множество всех элементов универсального множества U, не принадлежащих A.

**Алгоритм по решению задачи:**

1. Создаем универсальное множество Un мощностью 100;
2. Заполняем универсальное множество Un всеми элементами от 1 до 100;
3. Просим пользователя ввести мощность множества А (от 1 до 10);
4. Просим пользователя ввести мощность множества В (от 1 до 10);
5. Спрашиваем у пользователя, как задать множества: вручную или автоматически;
   1. Если пользователь выбрал задать множество вручную, то перейдем к пункту 6;
   2. Если пользователь выбрал задать множество автоматически, то перейдем к пункту 8;
6. Просим пользователя ввести значения множества A;
7. Просим пользователя ввести значения множества B;
8. Если выбрано автоматическое заполнение, заполняем множества А и В случайными числами от 1 до 100;
9. Находим объединение множеств:
   1. Создаем пустое множество M;
   2. Записываем все элементы множества A во множество M;
   3. Для каждого элемента множества B:
      1. Если элемент отсутствует в M, добавляем его в M;
   4. Выводим на экран все значения множества М;
10. Находим пересечение множеств:
    1. Создаем пустое множество Р;
    2. Для каждого элемента множества A:
       1. Если элемент присутствует в B, добавляем его в P;
    3. Выводим на экран значения множества Р;
11. Находим разность множеств:
    1. Находим разность множеств В\А:
       1. Создаем пустое множество R1;
       2. Для каждого элемента множества B:
          1. Если элемент отсутствует в A, добавляем его в R1;
       3. Выводим на экран множество R1;
    2. Находим разность множеств А\В:
       1. Создаем пустое множество R2;
       2. Для каждого элемента множества A:
          1. Если элемент отсутствует в B, добавляем его в R2;
       3. Выводим на экран множество R2;
12. Находим симметрическую разность множеств А и В:
    1. Создаем пустое множество S\_R;
    2. Для каждого элемента множества A:
       1. Если элемент отсутствует в B, добавляем его в S\_R;
    3. Для каждого элемента множества B:
       1. Если элемент отсутствует в A, добавляем его в S\_R;
    4. Выводим на экран значения множества S\_R;
13. Находим дополнение множеств:
    1. Дополнение множества A:
       1. Создаем пустое множество A\_complement;
       2. Для каждого элемента из Un:
          1. Если элемент отсутствует в A, добавляем его в A\_complement;
       3. Выводим на экран множество A\_complement;
    2. Дополнение множества B:
       1. Создаем пустое множество B\_complement;
       2. Для каждого элемента из Un:
          1. Если элемент отсутствует в B, добавляем его в B\_complement;
       3. Выводим на экран множество B\_complement;

**Вывод:**  
В ходе выполнения лабораторной работы был разработан алгоритм для выполнения базовых операций над множествами. Алгоритм позволяет пользователю вводить два множества и выполняет над ними операции объединения, пересечения, разности, симметрической разности и дополнения. Реализация данного алгоритма на языке программирования позволит эффективно работать с множествами и выполнять различные теоретико-множественные операции.