

# ЛАБОРАТОРНАЯ РАБОТА № 1

## МОДЕЛИРОВАНИЕ ОПЕРАЦИЙ НАД МНОЖЕСТВАМИ.

**Цель работы:** Изучить способы численного моделирования операций пересечения, дополнения и разности для множеств.

### Теоретические сведения

Под множеством  $M$  понимается совокупность некоторых объектов, которые будут называться *элементами множества*. Элементы множества сами могут являться множествами. Множество  $M$  может быть бесконечным или конечным, т.е. состоять из конечного числа элементов. Множество можно задать перечислением принадлежащих ему элементов или указанием свойств, которым элементы множества должны удовлетворять. Например, множество  $M$  арабских цифр можно задать: перечислением  $M=\{0,1,2,3,4,5,6,7,8,9\}$  или предикатом  $M=\{x \mid x - \text{арабская цифра}\}$ .

Над множествами можно выполнять различные операции, такие как *пересечение*, *объединение*, *дополнение*, *вычитание*, и др.

**Объединением** множеств  $A$  и  $B$  называется новое множество, состоящее из всех тех элементов, которые принадлежат хотя бы одному из множеств  $A$  или  $B$ .

$$A \cup B = \{x \mid x \in A \vee x \in B\}$$

**Пересечением** множеств  $A$  и  $B$  называется множество, состоящее из всех тех и только тех элементов, которые принадлежат одновременно как множеству  $A$ , так и множеству  $B$ .

$$A \cap B = \{x \mid x \in A \ \& \ x \in B\}$$

**Разностью** множеств  $A$  и  $B$  называется множество всех тех и только тех элементов  $A$ , которые не содержатся в  $B$ .

$$A \setminus B = \{x \mid x \in A \ \& \ x \notin B\}$$

### Задание на лабораторную работу:

Подготовить класс, реализующий пересечение, объединение и разности двух конечных множеств. С использованием данного класса реализовать оконное приложение для работы с двумя множествами.

Интерфейсная часть должна включать:

- полей ввода элементов для двух множеств;
- поля вывода результатов;
- управляющих кнопок для выполнения операций над множествами.

Множества должны быть представлены наборами однотипных данных. В текущей работе должны рассматриваться наборы целых чисел, а также произвольные наборы.

Для хранения множества целых чисел будем использовать массивы `int`, на произвольных наборов – массивы `String`.

Так как множества должны состоять из неповторяющихся элементов, то в классе должны быть реализованы методы для проверки на уникальность элементов:

```
bool isSet(int[] A)
```

```
bool isSet(String[] A)
```

Реализовать метод для исключения повторяющихся элементов:

```
int[] Remove(int[] A)
```

```
String[] Remove(String[] A)
```

Метод выполнения объединения двух множеств:

```
int[] Union(int[] A, int[] B)
```

```
String[] Union(String[] A, String[] B)
```

Метод выполнения пересечения двух множеств:

```
int[] Intersection(int[] A, int[] B)
```

```
String[] Intersection(String[] A, String[] B)
```

Метод выполнения разности двух множеств:

```
int[] Difference(int[] A, int[] B)
```

```
String[] Difference(String[] A, String[] B)
```