

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ
БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт математики и информационных систем
Факультет автоматики и вычислительной техники
Кафедра электронных вычислительных машин

Отчёт по лабораторной работе №1
по дисциплине
«Дискретная Математика»
«Работа с множествами.»

Разработал студент гр. ИВТб-1301-05-00 _____ /Черкасов А. А./
(подпись)

Проверил старший преподаватель _____ /Пахарева И. В./
(подпись)

Киров
2025

Цель работы

Цель работы: Научиться использовать множества на языке Pascal.

Задание

Требуется реализовать программу для выполнения заданных операций над множествами.

Вариант 27.

A	B	C	D	E	X	Y	K
\mathbb{R}	\mathbb{Q}	\mathbb{Z}	лат.	кир.	$A \cap B \cap C$	$E \cup D$	$X \Delta Y$

1. Программа должна позволять вводить множества с именами (A , B , C и т.д.) за счёт использования строки с заданным синтаксисом ($\langle \text{имя множества} \rangle = \langle \text{элемент 1} \rangle, \langle \text{элемента 2} \rangle, \dots$) или поэлементного ввода согласно критерию типа множества (по варианту). Множество может включать в себя до десяти элементов.
2. Программа должна позволять выполнять заданные операции (по вариантам) над введенными множествами с выводом результата.
3. В случае некорректного введения элемента множества должно появляться информационное сообщение об ошибке.
4. Определить мощность итогового множества K .

Решение

Рисунок 1.1 - Схема прямого счетчика.

Приложение А1

```
uses crt;

type
  TStringSet = set of string;

var
  A, B, C, D, E, X, Y, K: TStringSet; // Множества для хранения элементов
  choice: string; // Переменная для хранения выбора пользователя
  input, element: string; // Переменные для ввода и обработки элементов множества
  i: Integer; // Счетчик для циклов
  value: double; // Переменная для проверки числовых значений
  Power: Integer; // Переменная для хранения мощности множества K

procedure ClearAndShowMenu;
begin
  ClrScr;
  WriteLn('Меню:');
  WriteLn('1. Ввести множество A (R)');
  WriteLn('2. Ввести множество B (Q)');
  WriteLn('3. Ввести множество C (Z)');
  WriteLn('4. Ввести множество D (лат.)');
  WriteLn('5. Ввести множество E (рус.)');
  WriteLn('6. Выполнить операции над множествами');
  WriteLn('7. Выход');
  Write('Выберите действие: ');
end;

function IsRussianLetter(s: string): boolean;
var
  ch: Char;
```

```

    code: Integer;
begin
    if Length(s) = 1 then
    begin
        ch := s[1];
        code := Ord(ch);
        IsRussianLetter := ((code >= 1040) and (code <= 1103))
        or (code = Ord('ё')) or (code = Ord('Ё'));
    end
    else
        IsRussianLetter := False;
    end;

// input передается по ссылке, возвращаемое значение - элемент множества
function ExtractElement(var input: string): string;
var
    posComma, posSpace: Integer; // Позиции разделителей в строке
begin
    posComma := Pos(',', input);
    posSpace := Pos(' ', input);
    if (posComma = 0) or ((posSpace <> 0) and (posSpace < posComma)) then
        posComma := posSpace;
    if posComma = 0 then
        posComma := Length(input) + 1;
    Result := Trim(Copy(input, 1, posComma - 1));
    Delete(input, 1, posComma);
end;

procedure InputSet(var S: TStringSet; const setName: string; const criteria: string);
begin
    Write('Введите элементы множества ', setName, ' (', criteria, '): ');
    ReadLn(input);
    S := [];

```

```

i := 1;
while (i <= 10) and (input <> '') do
begin
    element := ExtractElement(input);
    case setName of
        'A':
            begin
                if TryStrToFloat(element, value) then
                    S += [element]
                else
                    begin
                        WriteLn('Ошибка: Множество A может содержать только действительные
числа.');

```

```

    end;
'D':
begin
    if (Length(element) = 1) and ((ord(element[1]) >= 65)
    and (ord(element[1]) <= 90)) or ((ord(element[1]) >= 97)
    and (ord(element[1]) <= 122)) then S += [element]
    else
    begin
        WriteLn('Ошибка: Множество D может содержать только латинские буквы.');
```

```

        WriteLn('Нажмите любую клавишу для продолжения...');
        ReadKey;
    end;
end;
'E':
begin
    if IsRussianLetter(element) then
        S += [element]
    else
    begin
        WriteLn('Ошибка: Множество E может содержать только русские буквы.');
```

```

        WriteLn('Нажмите любую клавишу для продолжения...');
        ReadKey;
    end;
end;
'C':
begin
    if TryStrToInt(element, i) and (i > 0) then
        S += [element]
    else
    begin
        WriteLn('Ошибка: Множество C может содержать только положительные
        целые числа.');
```

```

        WriteLn('Нажмите любую клавишу для продолжения...');
```

```

        ReadKey;
    end;
end;
end;
Inc(i);
end;
end;

begin
    repeat
        ClearAndShowMenu;
        ReadLn(choice);

        case choice of
            '1': InputSet(A, 'A', 'R - множество действительных чисел');
            '2': InputSet(B, 'B', 'Q - множество рациональных чисел');
            '3': InputSet(C, 'C', 'Z - множество целых чисел');
            '4': InputSet(D, 'D', 'лат. ');
            '5': InputSet(E, 'E', 'рус. ');
            '6':
                begin
                    X := A * B * C; // Пересечение множеств
                    Y := (E + D) - (E * D); // Симметрическая разность
                    K := X + Y; // Объединение

                    // Вывод результатов:
                    Write('X (A \cap B \cap C) = {');
                    var XArray := X.ToArray;
                    for i := Low(XArray) to High(XArray) do
                        begin
                            Write(XArray[i]);
                            if i < High(XArray) then
                                Write(', ');

```

```

end;
WriteLn('}');

Write('Y (E \triangle D) = {');
var YArray := Y.ToArray;
for i := Low(YArray) to High(YArray) do
begin
    Write(YArray[i]);
    if i < High(YArray) then
        Write(', ');
end;
WriteLn('}');

Write('K (X \cup Y) = {');
var KArray := K.ToArray;
for i := Low(KArray) to High(KArray) do
begin
    Write(KArray[i]);
    if i < High(KArray) then
        Write(', ');
end;
WriteLn('}');

Power := K.Count; // Подсчет мощности K
WriteLn('Мощность множества K: ', Power);

// Пауза перед перерисовкой меню
WriteLn('Нажмите любую клавишу для продолжения...');
ReadKey;
end;
'7': WriteLn('Программа завершена.');
```

else

```

WriteLn('Неверный выбор. Попробуйте снова.');
```



```
end;
```

```
until choice = '7';
```

```
end.
```

Вывод

В ходе работы удалось научиться использовать и проводить операции над множествами на языке Pascal.