Министерство образования Российской Федерации

Пермский Национальный Исследовательский Политехнический Университет

Кафедра «Информационные технологии и автоматизированные системы»

Лабораторная работа № 1

Тема: «Калькулятор множеств»

Выполнил:

Студент группы ИВТ-19-2б

Шеретов Марк Алексеевич

Принял:

Доцент кафедры ИТАС

А.Ф. Хабибулин

Пермь 2020

ПОСТАНОВКА ЗАДАЧИ

Разработать программу «Калькулятор» для работы со множествами чисел на любом из языков программирования.

АНАЛИЗ ЗАДАЧИ

На вход программа получает:

* Длину множества универсум (множество принимает вид: 1, 2, …, N );
* Длину множества А;
* Объекты множества А;
* Длину множества B;
* Объекты множества B;
* Длину множества C;
* Объекты множества C;
* Длину множества D;
* Объекты множества D;
* Количество скобок в конечном выражении;
* Содержание каждой из скобок (литералы обозначающие множества, объединённые различными идентификаторами операций);
* Идентификатор операции между скобок.

Программа способна воспринимать литералы верхнего и нижнего регистра. Идентификатор операций представлен в 2х различных типах:

* Сложение: «v» , «+»;
* Объединение: «^» , «\*»;
* Отрицание: «!» , «-».

На выход программа подаёт множество чисел, полученное в результате выполнения операций над исходными множествами, описанное в исходных данных.

КОД ПРОГРАММЫ НА ЯП C#:

using System;

using System.Collections.Generic;

using System.Linq;

namespace LogicalCalculator

{

class Program

{

static int n;

static List<int> U, A, B, C, D;

static IEnumerable<int> mn1, mn2;

static void Main(string[] args)

{

Run();

}

static void Run()

{

InputU();

InputA();

InputB();

InputC();

InputD();

parentheses();

}

static IEnumerable<int> Parse(string str)

{

bool flag, neg1, neg2, inter, un;

flag = neg1 = neg2 = inter = un = false;

for (int i = 0; i < str.Length; i++)

{

if (i == 0 && (str[i] == '!' || str[i] == '-')) neg1 = true;

else if (i != 0 && (str[i] == '!' || str[i] == '-')) neg2 = true;

else if (str[i] == 'A' || str[i] == 'a')

if (flag == false)

{

mn1 = A;

flag = true;

}

else mn2 = A;

else if (str[i] == 'B' || str[i] == 'b')

if (flag == false)

{

mn1 = B;

flag = true;

}

else mn2 = B;

else if (str[i] == 'C' || str[i] == 'c')

if (flag == false)

{

mn1 = C;

flag = true;

}

else mn2 = C;

else if (str[i] == 'D' || str[i] == 'd')

if (flag == false)

{

mn1 = D;

flag = true;

}

else mn2 = D;

else if (str[i] == '^') inter = true;

else if (str[i] == 'v') un = true;

}

if (neg1 == true)

{

mn1 = Exception(mn1);

if (inter == false && un == false)

return mn1;

}

if (neg2 == true)

{

mn2 = Exception(mn2);

if (inter == false && un == false)

return mn2;

}

if (inter == true) return (Intersection(mn1, mn2));

if (un == true) return (Union(mn1, mn2));

return mn1;

}

static IEnumerable<int> Exception(IEnumerable<int> a)

{

return U.Except(a);

}

static IEnumerable<int> Intersection(IEnumerable<int> a, IEnumerable<int> b)

{

return a.Intersect(b);

}

static IEnumerable<int> Union(IEnumerable<int> a, IEnumerable<int> b)

{

return a.Union(b);

}

static void InputU()

{

Console.Write("Количество элементов в универсуме: ");

n = Convert.ToInt32(Console.ReadLine());

U = new List<int>(n);

for (int i = 1; i <= n; i++)

U.Add(i);

}

static void InputA()

{

Console.Write("Количество элементов в A: ");

n = Convert.ToInt32(Console.ReadLine());

A = new List<int>(n);

for (int i = 0; i < n; i++)

A.Add(Convert.ToInt32(Console.ReadLine()));

}

static void InputB()

{

Console.Write("Количество элементов в B: ");

n = Convert.ToInt32(Console.ReadLine());

B = new List<int>(n);

for (int i = 0; i < n; i++)

B.Add(Convert.ToInt32(Console.ReadLine()));

}

static void InputC()

{

Console.Write("Количество элементов в C: ");

n = Convert.ToInt32(Console.ReadLine());

C = new List<int>(n);

for (int i = 0; i < n; i++)

C.Add(Convert.ToInt32(Console.ReadLine()));

}

static void InputD()

{

Console.Write("Количество элементов в D: ");

n = Convert.ToInt32(Console.ReadLine());

D = new List<int>(n);

for (int i = 0; i < n; i++)

B.Add(Convert.ToInt32(Console.ReadLine()));

}

static void parentheses()

{

Console.Write("Количество скобок: ");

n = Convert.ToInt32(Console.ReadLine());

IEnumerable<int>[] parentheses = new IEnumerable<int> [n];

for (int i = 0; i < parentheses.Length; i++)

{

Console.Write("Скобка({0}): ", i+1);

string st = Console.ReadLine();

parentheses[i] = Parse(st);

}

if (n != 1)

{

Console.Write("Операция между скобками: ");

string buf = Console.ReadLine();

if (buf == "^" || buf == "\*")

for (int i = 0; i < (parentheses.Length - 1); i++)

parentheses[i + 1] = Intersection(parentheses[i], parentheses[i + 1]);

if (buf == "v" || buf == "+")

for (int i = 0; i < (parentheses.Length - 1); i++)

parentheses[i + 1] = Union(parentheses[i], parentheses[i + 1]);

}

Console.Write("Ответ: ");

foreach (int element in parentheses[n - 1])

Console.Write($"{element} ");

}

}

}