|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** ***М-КФ «Машиностроительный»***

**КАФЕДРА** \_***М10-КФ «Высшая математика и физика»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

**ЛАБОРАТОРНАЯ РАБОТА №1**

**«Операции над множествами»**

**ДИСЦИПЛИНА: «Дискретная математика»**

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИУК4-32Б | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Карельский М.К. )  (Подпись) |
| Проверил: доцент кафедры М10-КФ | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Булычев В.А. )  (Подпись) |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | | |
|  | - Балльная оценка:  - Оценка: | |

Калуга, 2021

**Цель:** реализация основных операций над множествами (пересечение, объединение, разность, симметрическая разность, дополнение), заданных в виде бинарных массивов.

**Задание:** в качестве *универсума* используется множество заглавных букв латинского алфавита (26 букв).

1. Прочитайте из файла «data.txt» два бинарных массива *A* и *B*, задающих множества *A* и *B* для вашего варианта, и напечатайте соответствующие им множества букв. Найдите их мощность.
2. Используя оператор цикла, арифметические операции и условный оператор, получите бинарные массивы, соответствующие пересечению, объединению, разностям, симметрической разности и дополнениям множеств A и B.
3. Для каждого из 7-ми массивов пункта 2 напечатайте соответствующие множества букв и найдите их мощность.
4. В качестве **контрольного значения** выведите сумму мощностей 9-ти множеств, полученных в предыдущих пунктах.

**Вариант 10**

**Листинг:**

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

void main()

{

setlocale(LC\_ALL, "Russian");

string A{};

string B{};

ifstream fileRead("data.txt");

fileRead >> A;

fileRead >> B;

fileRead.close();

cout << "Множество Бинарный массив Мощность Буквы \n";

size\_t powerOfA = 0;

string lettersOfA{};

for (size\_t i{}; i < A.length(); ++i)

{

if (A[i] == '1')

{

++powerOfA;

lettersOfA.push\_back(65 + i);

}

}

cout << "A " << A << " " << powerOfA << " ";

if (powerOfA < 10)

{

cout << " ";

}

cout << lettersOfA << "\n";

size\_t powerOfB = 0;

string lettersOfB{};

for (size\_t i{}; i < B.length(); ++i)

{

if (B[i] == '1')

{

++powerOfB;

lettersOfB.push\_back(65 + i);

}

}

cout << "B " << B << " " << powerOfB << " ";

if (powerOfB < 10)

{

cout << " ";

}

cout << lettersOfB << "\n";

size\_t powerOfAAndB = 0;

string letterOfAAndB{};

cout << "A & B ";

for (size\_t i{}; i < A.length(); ++i)

{

if (A[i] == '1' && B[i] == '1')

{

++powerOfAAndB;

letterOfAAndB.push\_back(65 + i);

cout << "1";

}

else

{

cout << "0";

}

}

cout << " " << powerOfAAndB << " ";

if (powerOfAAndB < 10)

{

cout << " ";

}

cout << letterOfAAndB << "\n";

size\_t powerOfAOrB = 0;

string letterOfAOrB{};

cout << "A U B ";

for (size\_t i{}; i < A.length(); ++i)

{

if (A[i] == '1' || B[i] == '1')

{

++powerOfAOrB;

letterOfAOrB.push\_back(65 + i);

cout << "1";

}

else

{

cout << "0";

}

}

cout << " " << powerOfAOrB << " ";

if (powerOfAOrB < 10)

{

cout << " ";

}

cout << letterOfAOrB << "\n";

size\_t powerOfAMinusB = 0;

string letterOfAMinusB{};

cout << "A \\ B ";

for (size\_t i{}; i < A.length(); ++i)

{

if (A[i] == '1' && B[i] == '0')

{

++powerOfAMinusB;

letterOfAMinusB.push\_back(65 + i);

cout << "1";

}

else

{

cout << "0";

}

}

cout << " " << powerOfAMinusB << " ";

if (powerOfAMinusB < 10)

{

cout << " ";

}

cout << letterOfAMinusB << "\n";

size\_t powerOfBMinusA = 0;

string letterOfBMinusA{};

cout << "B \\ A ";

for (size\_t i{}; i < A.length(); ++i)

{

if (A[i] == '0' && B[i] == '1')

{

++powerOfBMinusA;

letterOfBMinusA.push\_back(65 + i);

cout << "1";

}

else

{

cout << "0";

}

}

cout << " " << powerOfBMinusA << " ";

if (powerOfBMinusA < 10)

{

cout << " ";

}

cout << letterOfBMinusA << "\n";

size\_t powerOfADeltaB = 0;

string letterOfADeltaB{};

cout << "B d A ";

for (size\_t i{}; i < A.length(); ++i)

{

if (A[i] == '1' && B[i] == '0' ||

A[i] == '0' && B[i] == '1')

{

++powerOfADeltaB;

letterOfADeltaB.push\_back(65 + i);

cout << "1";

}

else

{

cout << "0";

}

}

cout << " " << powerOfADeltaB << " ";

if (powerOfADeltaB < 10)

{

cout << " ";

}

cout << letterOfADeltaB << "\n";

size\_t powerOfNotA = 0;

string letterOfNotA{};

cout << "- A ";

for (size\_t i{}; i < A.length(); ++i)

{

if (A[i] == '0')

{

++powerOfNotA;

letterOfNotA.push\_back(65 + i);

cout << "1";

}

else

{

cout << "0";

}

}

cout << " " << powerOfNotA << " ";

if (powerOfNotA < 10)

{

cout << " ";

}

cout << letterOfNotA << "\n";

size\_t powerOfNotB = 0;

string letterOfNotB{};

cout << "- B ";

for (size\_t i{}; i < B.length(); ++i)

{

if (B[i] == '0')

{

++powerOfNotB;

letterOfNotB.push\_back(65 + i);

cout << "1";

}

else

{

cout << "0";

}

}

cout << " " << powerOfNotB << " ";

if (powerOfNotB < 10)

{

cout << " ";

}

cout << letterOfNotB << "\n";

cout << "Контрольное значение: " << powerOfA + powerOfB +

powerOfAAndB + powerOfAOrB + powerOfAMinusB + powerOfBMinusA +

powerOfADeltaB + powerOfNotA + powerOfNotB << "\n";

}

**Результат:**

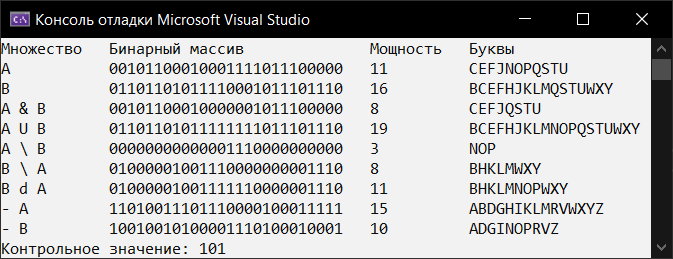


Рис. 1. Результат работы

**Вывод:** в ходе выполнения лабораторной работы составлена программа, реализующая операции над множествами, заданными в виде бинарных массивов.