

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту

Лабораторна робота

з дисципліни
«Дискретна математика»

Виконав:
студент групи КН-109
Гавришків Олексій
Викладач:
Бойко Н. І.

Львів – 2018 р.

Тема: Моделювання основних операцій для числових множин.

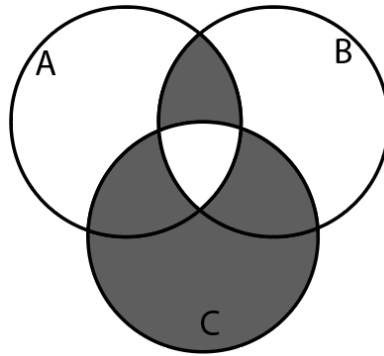
Мета роботи: Ознайомитись на практиці із основними поняттями теорії множин, навчитись будувати діаграми Ейлера-Венна операцій над множинами, використовувати закони алгебри множин, освоїти принцип включень-виключень для двох і трьох множин та комп'ютерне подання множин.

Варіант 1

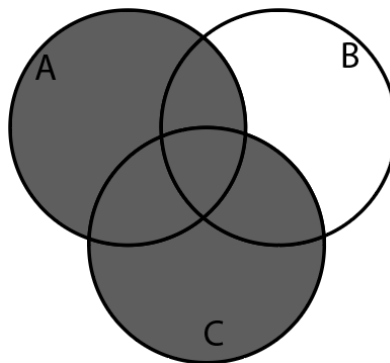
- Для скінченних множин $A = \{1, 2, 3, 4, 5, 6, 7\}$, $B = \{4, 5, 6, 7, 8, 9, 10\}$, $C = \{2, 4, 6, 8, 10\}$ та універсума $U = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ знайти множину, яку задано за допомогою операцій: а) $(A \cap B) \cup C$; б) $(A \cup C) \setminus B$. Розв'язати, використовуючи комп'ютерне подання множин.
а) $(A \cap B) \cup C$
 $(A \cap B) = \{4, 5, 6, 7\}$
 $(A \cap B) \cup C = \{2, 4, 5, 6, 7, 8, 10\}$
б) $(A \cup C) \setminus B$
 $(A \cup C) = \{1, 2, 3, 4, 5, 6, 7, 8, 10\}$
 $(A \cup C) \setminus B = \{1, 2, 3\}$
Комп'ютерне подання: $(A \cup C) \setminus B = \{1, 1, 1, 0, 0, 0, 0, 0, 0, 0\}$
- На множинах задачі 1 побудувати булеан множини $\overline{(B \setminus A) \cup C}$. Знайти його потужність.
 $(B \setminus A) = \{4, 5, 6, 7\}$
 $(B \setminus A) \cup C = \{2, 4, 5, 6, 7, 8, 10\}$
 $D = \overline{(B \setminus A) \cup C} = \{1, 3, 9\}$
 $P(D) = \{\{\emptyset\}, \{1, 3, 9\}, \{1\}, \{3\}, \{9\}, \{1, 3\}, \{1, 9\}, \{3, 9\}\}$
 $|\overline{(B \setminus A) \cup C}| = 3$
- Нехай маємо множини: N – множина натуральних чисел, Z – множина цілих чисел, Q – множина раціональних чисел, R – множина дійсних чисел; A, B, C – будь-які множини. Перевірити які твердження є вірними (в останній задачі у випадку невірною твердження достатньо навести контрприклад, якщо твердження вірне – навести доведення):
а) $\{2, 3\} \in \{\{1, 2\}, 2, 3\}$; б) $Q \subset R$; в) $Q \cup N \subset N$; г) $N \cap Z \subset Z \cap Q$; д) якщо $C \subset \overline{A} \cup \overline{B}$, то $A \cap B \subset \overline{C}$
а) $\{\{1, 2\}, 2, 3\} = \{1, 2, 3\}$; $\{2, 3\} \in \{1, 2, 3\}$, отже твердження $\{2, 3\} \in \{\{1, 2\}, 2, 3\}$ **правильне**.
б) Оскільки всі раціональні числа входять до множини дійсних чисел, то твердження $Q \subset R$ **правильне**.
в) Твердження $Q \cup N \subset N$ **не правильне**, оскільки $N \subset Q$.
Контрприклад: $Q \cup N \subseteq N$.
г) $N \cap Z$ – це всі натуральні числа, а $Z \cap Q$ – це всі цілі числа. Тобто вираз можна спростити до вигляду $N \subset Z$. Всі натуральні числа належать цілим, отже твердження $N \cap Z \subset Z \cap Q$ **правильне**.
д) За законами де Моргана $\overline{A} \cup \overline{B} = \overline{A \cap B}$. Але умова є **не правильна**.
Контрприклад: $C \subset A \cap B$

4. Логічним методом довести тотожність: $A \cap (B \setminus C) = (A \cap B) \setminus (A \cap C)$.
За законом дистрибутивності розкладаємо ліву частину:
 $A \cap (B \setminus C) = (A \cap B) \setminus (A \cap C)$. Оскільки $(A \cap B) \setminus (A \cap C) = (A \cap B) \setminus (A \cap C)$, то вирази тотожні.
5. Зобразити на діаграмі Ейлера-Венна множину: $((A \cap B) \Delta C) \setminus (A \cup C)$.
Розглянемо окремо вирази (1) $((A \cap B) \Delta C)$ і (2) $(A \cup C)$.

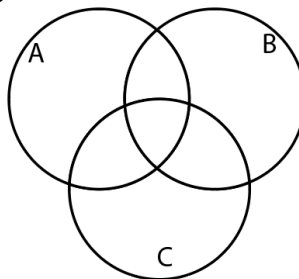
На діаграмі Ейлера-Венна перший вираз буде мати наступний вигляд:



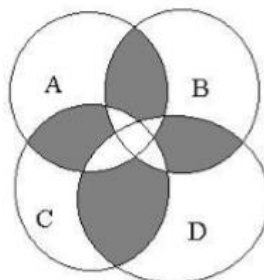
Другий вираз:



При відніманні від першого виразу другого ми отримаємо пусту множину, отже $((A \cap B) \Delta C) \setminus (A \cup C)$ буде мати вигляд:



6. Множину зображено на діаграмі. Записати її за допомогою операцій.



Відповідь: $((A \cap C) \cup (B \cap D)) \Delta ((A \cap B) \cup (C \cap D))$

7. Спростити вигляд множини, яка задана за допомогою операцій, застосовуючи закони алгебри множин (у відповідь множини можуть входити не більше одного разу): $\overline{A \cap B \cap C} \cup (A \cap B) \cup \bar{C}$

$$\begin{aligned} \overline{A \cap B \cap C} \cup (A \cap B) \cup \bar{C} &= \bar{A} \cup \bar{B} \cup \bar{C} \cup (A \cap B) \cup \bar{C} = \\ &= \bar{A} \cup \bar{B} \cup \bar{C} \cup (A \cap B) = (A \cap B) \cup (A \cap B) \cup \bar{C} = \\ &= (A \cap B) \cup \bar{C} \end{aligned}$$

8. Зі 100 студентів англійську мову знають 28 студентів, німецьку – 30, французьку – 42, англійську та німецьку – 8, англійську та 15 французьку – 10, німецьку та французьку – 5, всі 3 мови знають 3 студента. Скільки студентів не знають жодну з трьох мов?

$|A \cap B \cap C| = 3$ – всі мови

$|A| = 28$ англійську

$|B| = 30$ німецьку

$|C| = 42$ французьку

$|A \cap B| = 8$ англ. і нім.

$|A \cap C| = 10$ англ. і франц.

$|B \cap C| = 5$ франц. і нім.

$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap C| - |A \cap B| - |B \cap C| +$

$+ |A \cap B \cap C| = 28 + 30 + 42 - 8 - 10 - 5 + 3 = 80$ знають хоча б одну мову.

Якщо всіх студентів 100, то ті, що не знають жодну з мов $= 100 - 80 = 20$

Додаток 2

Ввести з клавіатури дві множини цілих чисел. Реалізувати операції об'єднання та симетричної різниці над цими множинами. Вивести на екран новоутворені множини. Знайти програмно їх потужність.

Код програми на c++:

```
#include <iostream>
using namespace std;

void arr_print(int values[100], int n) { // Виведення заданого масиву
    cout << "{";
    for (int i = 0; i < n; i++) {
        if (i != 0) {
            cout << ", ";
        }
        cout << values[i];
    }
    cout << "}\n";
}

int clean(int values[100], int n) { // Очищення множини від однакових елементів! Повертає потужність
    int count = 0;
    for (int i = 0; i < n; i++) {
        int b = 1;
        for (int j = 0; j < i; j++) {
```

```

        if (values[i] == values[j]) {
            count++;
            b = 0;
            break;
        }
    }
    if (b) {
        values[i - count] = values[i];
    }

}
return (n - count);
}

int intersection(int arr1[100], int arr2[100], int arr3[100], int count1,
int count2) { // Перетин двох множин arr1 i arr2
    count1 = clean(arr1, count1);
    count2 = clean(arr2, count2);
    int index = 0;
    for (int i = 0; i < count1; i++) {
        for (int j = 0; j < count2; j++) {
            if (arr1[i] == arr2[j]) {
                arr3[index] = arr1[i];
                index++;
                break;
            }
        }
    }
    return index;
}

int merger(int arr1[100], int arr2[100], int arr3[100], int count1, int
count2) { // Об'єднання двох множин arr1 i arr2
    count1 = clean(arr1, count1);
    count2 = clean(arr2, count2);
    int count3 = count1 + count2;
    for (int i = 0; i < count1; i++) {
        arr3[i] = arr1[i];
    }
    for (int i = count1; i < count3; i++) {
        arr3[i] = arr2[i - count1];
    }
    count3 = clean(arr3, count3);
    return count3;
}

int rzn(int arr1[100], int arr2[100], int arr3[100], int count1, int
count2) { // Різниця двох множин
    int is, count = 0;
    count1 = clean(arr1, count1);
    count2 = clean(arr2, count2);
    for (int i = 0; i < count1; i++) {
        is = 0;
        for (int j = 0; j < count2; j++) {
            if (arr1[i] == arr2[j]) {
                is = 1;
                count++;
                break;
            }
        }
        if (!is) {
            arr3[i - count] = arr1[i];
        }
    }
}

```

```

    }
    return (count1 - count);
}
int sym_rizn(int arr1[100], int arr2[100], int arr3[100], int count1, int
count2) { // Симетрична різниця
    int arr_merge[100], arr_intersection[100];
    count1 = clean(arr1, count1);
    count2 = clean(arr2, count2);
    return
rizn(arr_merge, arr_intersection, arr3, merger(arr1, arr2, arr_merge, count1, co
unt2), intersection(arr1, arr2, arr_intersection, count1, count2));
}
int main() {
    setlocale(LC_ALL, "Ukrainian");
    int A[100], B[100], AwithB[100], cA, cB, pot;
    cout << "Введіть к-сть елементів множини A: ";
    cin >> cA;
    cout << "Введіть елементи множини A\n";
    for (int i = 0; i < cA; i++) {
        cout << i + 1 << ". ";
        cin >> A[i];
    }
    cout << "Введіть к-сть елементів множини B: ";
    cin >> cB;
    cout << "Введіть елементи множини B\n";
    for (int i = 0; i < cB; i++) {
        cout << i + 1 << ". ";
        cin >> B[i];
    }
    cout << "Об'єднання: ";
    pot = merger(A, B, AwithB, cA, cB);
    arr_print(AwithB, pot);
    cout << "Потужність: " << pot << endl;
    cout << "Симетрична різниця: ";
    pot = sym_rizn(A, B, AwithB, cA, cB);
    arr_print(AwithB, pot);
    cout << "Потужність: " << pot << endl;
    system("pause");
    return 0;
}

```

Результат виконання програми:

```

Введіть к-сть елементів множини A: 5
Введіть елементи множини A
1. 1
2. 2
3. 3
4. 4
5. 5
Введіть к-сть елементів множини B: 7
Введіть елементи множини B
1. 4
2. 5
3. 6
4. 7
5. 8
6. 9
7. 10
Об'єднання: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
Потужність: 10
Симетрична різниця: {1, 2, 3, 6, 7, 8, 9, 10}
Потужність: 8

```

Висновок: На цій лабораторній роботі я ознайомився з основними поняттями теорії множин, навчився будувати діаграми Ейлера-Венна операцій над множинами, використовувати закони алгебри множин, освоїв принцип включень-виключень для двох і трьох множин.