МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Інститут комп'ютерних наук та інформаційних технологій

Кафедра систем штучного інтелекту

Лабораторна робота №2
з курсу “Дискретна математика ”

Виконав:
ст. гр. КН-110
Холод Ігор

Викладач:
Мельникова Н.І.

Львів – 2018

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.

Теоретичні відомості:

Множина – це сукупність об’єктів, які називають елементами.

Кажуть, що множина A є **підмножиною** множини S (цей факт позначають $A \subseteq S$, де $\subseteq$ – знак нестрогого включення), якщо кожен її
елемент автоматично є елементом множини S. Досить часто при цьому кажуть, що множина A міститься в множині S.

Якщо $A \subseteq S$ і $S \neq A$, то A називають **власною** (**строгою**, **істинною**) **підмножиною** S (позначають $A \subset S$, де $\subset$ – знак строгого включення).

Дві множини A та S називаються **рівними**, якщо вони складаються з однакових елементів. У цьому випадку пишуть A=S.

Якщо розглядувані множини є підмножинами деякої множини, то її називають **універсумом** або **універсальною** множиною і позначають літерою U (зауважимо, що універсальна множина існує не у всіх випадках).

Множини як об’єкти можуть бути елементами інших множин, Множину, елементами якої є множини, інколи називають **сімейством**.

Множину, елементами якої є всі підмножини множини A і тільки вони (включно з порожньою множиною та самою множиною A), називають **булеаном** або **множиною-степенем** множини A і позначають P(A).

Потужністю скінченної множини A називають число її елементів, позначають $|A|$.

Множина, яка не має жодного елемента, називається **порожньою** і позначається $\varnothing$.

Вважається, що порожня множина є підмножиною будь-якої множини, а також A ⊂ A.
Множина всіх підмножин множини A називається булеаном і позначається P(A). Потужність скінченної множини дорівнює кількості її елементів, позначається $|A|$. Потужність порожньої множини дорівнює 0.
Якщо $|A| = n$, то $|P(A)| = 2^n$

Варіант 13.

Завдання 1. Для даних 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 універсуму $U = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ знайти множину, яку задано за допомогою операцій: а) $A \cap (B \cup C)$; б) $\overline{B \triangle C}$. Розв'язати, використовуючи комп'ютерне подання множин.

$A = \{1, 2, 3, 4, 5, 6, 7\} = 1111111000$
$B = \{4, 5, 6, 7, 8, 9, 10\} = 0001111111$
$C = \{2, 4, 6, 8, 10\} = 0101010101$
а) $A \cap (B \cup C) = 1111111000 \cap 0101111111 = 0101111000 = \{2, 4, 5, 6, 7\}$
б) $\overline{B \triangle C} = 0000101010 \cup 0100000000 = 0100101010 = \{2, 5, 7, 9\}$

Завдання 2. На множинах задачі 1 побудувати булеан множини $C \setminus (B \setminus \overline{C}) \cap A$.Знайти його потужність.

$\overline{C} = \{1, 3, 5, 7, 9\}$;
$B \setminus \overline{C} = \{4, 5, 6, 7, 8, 9, 10\} \setminus \{1, 3, 5, 7, 9\} = \{4, 6, 8, 10\}$;
$(B \setminus \overline{C}) \cap A = \{4, 6, 8, 10\} \cap \{1, 2, 3, 4, 5, 6, 7\} = \{4, 6\}$;
Нехай $K = C \setminus (B \setminus \overline{C}) \cap A = \{2, 4, 6, 8, 10\} \setminus \{4, 6\} = \{2, 8, 10\}$
$P(A) = \{\{\varnothing\}, \{2\}, \{8\}, \{10\}, \{2, 8\}, \{2, 10\}, \{8, 10\}, \{2, 8, 10\}\}$
$|P(A)| = 2^3 = 8$

Завдання 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 невірного твердження достатньо навести контрприклад, якщо твердження вірне − навести доведення):

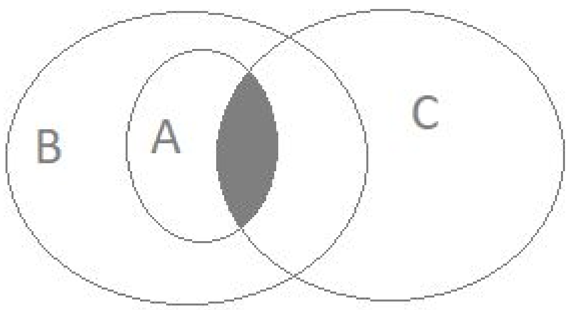
а) $\varnothing \in \{1, 2, 3\}$ - вірно, порожня множина є підмножиною будь-якої множини

б) $Z \subset R$ - вірно, цілі числа є дійсними

в) $Q \cup Z = Q$ - вірно, оскільки цілі числа є також і раціональними, то об'єднання цілих і раціональних чисел є раціональними числами

г) $R \setminus Z \subset R \setminus N$ - вірно, оскільки множина натуральних чисел є підмножиною множини дійсних чисел.

д) Якщо $A \subset B$, то $A \cap C \subset B \cap C$



Якщо $A \subset B$, то будь-яка множина, утворена перетином A і C буде міститись і в B, і в C, тобто в їх перетині $B \cap C$.

Завдання 4. Логічним методом довести тотожність: $\overline{A \cup B} \cap A = \varnothing$

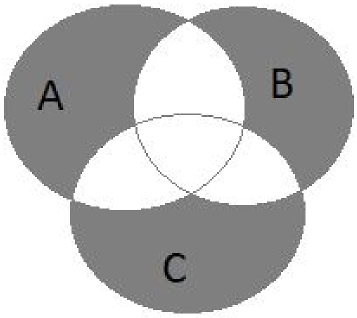
$\overline{A \cup B} \cap A \Rightarrow 1 \Rightarrow \overline{A} \cap \overline{B} \cap A \Rightarrow 2 \Rightarrow \varnothing \cap \overline{B} \Rightarrow 2 \Rightarrow \varnothing$

1. За законом де Моргана
2. За законом доповнення

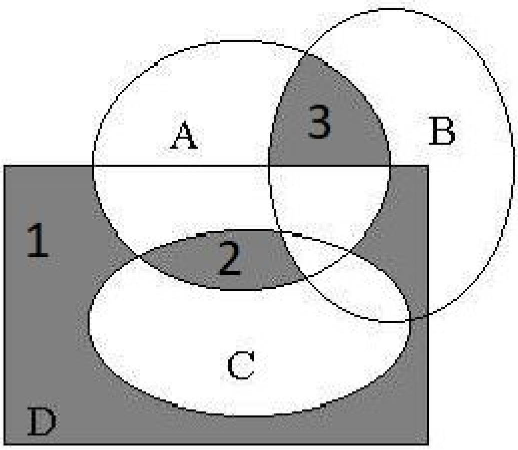
Отже, $\overline{A \cup B} \cap A = \varnothing$

Завдання 5. Зобразити на діаграмі Ейлера-Венна множину:

$(B \cup C) \Delta A \setminus (B \cap C)$



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



1. $D\setminus(A\cup C)$
2. $(A\cap C)\setminus B$
3. $(A\cap B)\setminus D$

Отже, готовий вигляд:

$(D\setminus(A\cup C))\cup((A\cap C)\setminus B)\cup((A\cap B)\setminus D)$

Завдання 7. Спростити вигляд множини, яка задана за допомогою операцій, застосовуючи закони алгебри множин (у відповідь множини можуть входити не більше одного разу):

$(A\cap\overline{B})\cup(A\cap B\cap C)\cup\overline{A\cap C}$ ⇒ за законом дистрибутивності

⇒ $A\cap(\overline{B}\cup(B\cap C))\cup\overline{A\cap C}$ ⇒ за законом дистрибутивності

⇒ $A\cap((\overline{B}\cup B)\cap(\overline{B}\cup C))\cup\overline{A\cap C}$ ⇒ за законом доповнення

⇒ $A\cap(U\cap(\overline{B}\cup C))\cup\overline{A\cap C}$ ⇒ за законом доповнення

⇒ $A\cap(\overline{B}\cup C)\cup\overline{A\cap C}$ ⇒ за законом де Моргана

⇒ $A\cap(\overline{B}\cup C)\cup\overline{A}\cup\overline{C}$ ⇒ за законом дистрибутивності

$\Rightarrow (A \cap \overline{B}) \cup (A \cap C) \cup \overline{A} \cup \overline{C}$ ⇒ за законом асоціативності

$\Rightarrow ((A \cap \overline{B}) \cup \overline{A}) \cup ((A \cap C) \cup \overline{C})$ ⇒ за законом дистрибутивності

$\Rightarrow ((A \cup \overline{A}) \cap (\overline{B} \cup \overline{A})) \cup ((A \cup \overline{C}) \cap (C \cup \overline{C}))$ ⇒ за законом доповнення

$\Rightarrow (U \cap (\overline{B} \cup \overline{A})) \cup (U \cap (A \cup \overline{C}))$ ⇒ за законом доповнення

$\Rightarrow (\overline{B} \cup \overline{A}) \cup (A \cup \overline{C})$ ⇒ за законом асоціативності

$\Rightarrow \overline{B} \cup \overline{A} \cup A \cup \overline{C}$ ⇒ за законом доповнення

$\Rightarrow \overline{B} \cup U \cup \overline{C}$ ⇒ за законом доповнення

$\Rightarrow \overline{B} \cup \overline{C}$

Завдання 8. Зі 100 студентів англійську мову вивчають 28 студентів, німецьку – 30, французьку – 42, англійську і французьку – 10, англійську і німецьку – 8, німецьку і французьку – 5, всі 3 мови студіюють троє. Скільки студентів не вивчають жодної із цих трьох мов?

Нехай:

A - вивчають англійську мову

B - вивчають німецьку мову

C - вивчають німецьку мову

Тоді: $|A| = 28$; $|B| = 30$; $|C| = 42$; $|A \cap C| = 10$; $|A \cap B| = 8$; $|B \cap C| = 5$; $|A \cap B \cap C| = 3$

$|A \cup B \cup C|$ - кількість студентів, які вивчають хоч 1 мову. Тоді кількість студентів, які не вивчають жодної мови $= 100 - |A \cup B \cup C|$.

За формулою включень-виключень:

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

$= 28 + 30 + 42 - 8 - 10 - 5 + 3 = 80$

$100 - |A \cup B \cup C| = 20$

Відповідь: 20 учнів не вивчають жодної мови.

Додаток до завдання 1.

Код програми:

```
#include <stdio.h>
#include <math.h>

int main()
{
	int U[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}; //Universum
	int n; //size of array

	//Inputting array's size
	printf("Input the size of your array (<10): ");
	scanf("%d", &n);

	int A[n]; //Starter array

	//Inputting array
	printf("Input your array (!!!ONLY INTEGER NUMBERS!!!):\n");
	for (int i = 0; i < n; i++)
	{
	printf("a[%d]=", i);
	if(((scanf("%d", &A[i]))==0)||(A[i]<=0)||(A[i]>10))
	{
	scanf("%*[^\n]");
	printf("Error. Try again. (Number has to be between 1 and 10)\n");
	i--;
	}
	}

	int Ab[10]; //Binary form array of A
	int Ad[10-n]; //Additional array for A
	int count = 0; //Counter for A

	//A --> Binary A
	for (int i = 0; i < 10; i++)
```

```
{
if (A[count] == U[i])
{
Ab[i] = 1;
count++;
}
else
Ab[i] = 0;
}

int Abd[10]; //Binary form of additional array

//Creating binary additional
for (int i = 0; i < 10; i++)
{
if (Ab[i] == 0)
Abd[i] = 1;
else
Abd[i] = 0;
}

//Printing your array
printf("Your array: {");
for (int i = 0; i < n; i++)
{
if (i != n-1)
printf("%d, ", A[i]);
else
printf("%d}", A[i]);
}

//Printing binary array
printf("\nBinary form of your array: ");
for (int i = 0; i < 10; i++)
printf ("%d", Ab[i]);

//Printing binary additional array
printf ("\nBinary form of additional array: ");
for (int i = 0; i < 10; i++)
printf ("%d", Abd[i]);

//Binary additional --> Additional
for (int i = 0, k = 0; i < 10; i++)
if (Abd[i] == 1)
{
Ad[k] = U[i];
```

```
        k++;
        }

        //Printing additional array
        printf("\nAdditional array: {");
        for (int i=0; i<10-n; i++)
        {
        if (i < 9-n)
        printf("%d, ", Ad[i]);
        else
        printf("%d}\n", Ad[i]);
        }

        int p = pow(2, n); //Power of boolean array

        //Printing the bolean
        printf("Bolean:\n");
        for (int i = 0; i < p; i++)
        {
        printf ("{ ");
        for (int k = 0; k < n; k++)
        {
                if (i & (1 << k))
                {
                printf ("%i ", A[k]);
                }
        }
        printf("}\n");
        }

        return 0;
}
```

Результат виконання:

```
Your array: {2, 4, 6, 8, 10}
Binary form of your array: 0101010101
Binary form of additional array: 1010101010
Additional array: {1, 3, 5, 7, 9}
Bolean:
{ }
{ 2 }
{ 4 }
{ 2 4 }
{ 6 }
{ 2 6 }
{ 4 6 }
{ 2 4 6 }
{ 8 }
{ 2 8 }
{ 4 8 }
{ 2 4 8 }
{ 6 8 }
{ 2 6 8 }
{ 4 6 8 }
{ 2 4 6 8 }
{ 10 }
{ 2 10 }
{ 4 10 }
{ 2 4 10 }
{ 6 10 }
{ 2 6 10 }
{ 4 6 10 }
{ 2 4 6 10 }
```

```
{ 8 10 }
{ 2 8 10 }
{ 4 8 10 }
{ 2 4 8 10 }
{ 6 8 10 }
{ 2 6 8 10 }
{ 4 6 8 10 }
{ 2 4 6 8 10 }
```

Висновок: я ознайомився з основними поняттями теорії множин, навчився на практиці застосовувати діаграми Ейлера-Венна, будувати їх. Також я навчився програмно реалізовувати деякі основні операції над множинами, такі як знаходження доповнення і побудова булеану якоїсь множини.