Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Дискретна математика

Лабораторна робота №1

«Множини: основні властивості та операції над ними, діаграми Венна».

Виконав: студент групи ІВ-91

 Степанюк Р. В.

 Залікова книжка № \_\_9127\_\_\_

Перевірив Новотарський М. А.

Київ

2020 р.

**Мета:** вивчити основні аксіоми, закони і теореми теорії множин, навчитисязастосовувати їх на практиці. Обчислити логічний вираз шляхом послідовного застосування операцій над множинами.

**Загальне завдання**

А) Вибрати номер Z індивідуального варіанта відповідно до виразу:

Z=(i+G%60)%30+1, де і – номер у списку групи, G – числова складова назви

групи, М-символьна складова назви групи.

Програма обчислення варіанта:

G = 91

N = 1

M = **"ІО"**

print(**"Моя група: ",** M **+ "-"**, G)

print(**"Мій номер у групі:"**, N)

if M==**"ІО":** N += 2

print(**"Мій варіант:"**, (N + G % 60) % 30 + 1)

Ця програма повинна входити в код лабораторної роботи.

Б) Максимально спростити логічний вираз. Для спрощення використати

тотожності алгебри множин та визначення логічних операцій. Спрощення є

максимальним, якщо формула містить тільки одне входження кожної множини.

В) Створити блок-схему послідовності обчислення початкового логічного

виразу.

Г) Створити блок-схему послідовності обчислення спрощеного логічного

виразу.

Д) З використанням блок-схем створити проект, який містить модуль з

функцією для обчислення початкового виразу (**1**) та модуль з функцією для

обчислення спрощеного виразу.

Е) З використанням блок-схем, заданих у лабораторній роботі, створити

модуль з функцією для виконання логічної операції (**2**), вибраної відповідно до

варіанта.

Ж) В основному файлі виконати перевірку правильності спрощення виразу з

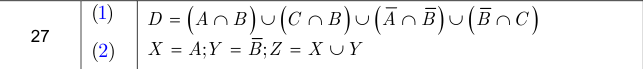
виводом відповідного повідомлення.

З) Як елементи множин можуть бути використані числа від 0 до 255.

І) Лабораторну роботу виконувати з застосуванням мови Python та

бібліотеки tkinter

**Індивідуальне завдання**



**Короткі теоретичні відомості по темі, які відображають правила виконання**

**логічних операцій, застосованих при виконанні лабораторної роботи**

Види операцій. Операції, які виконують над однією множиною, називають унарними. Операції, які виконують над двома множинами, називають бінарними. Прикладом унарної операції є знаходження доповнення. Прикладами бінарних операцій є об'єднання, перетин, різниця, симетрична різниця.

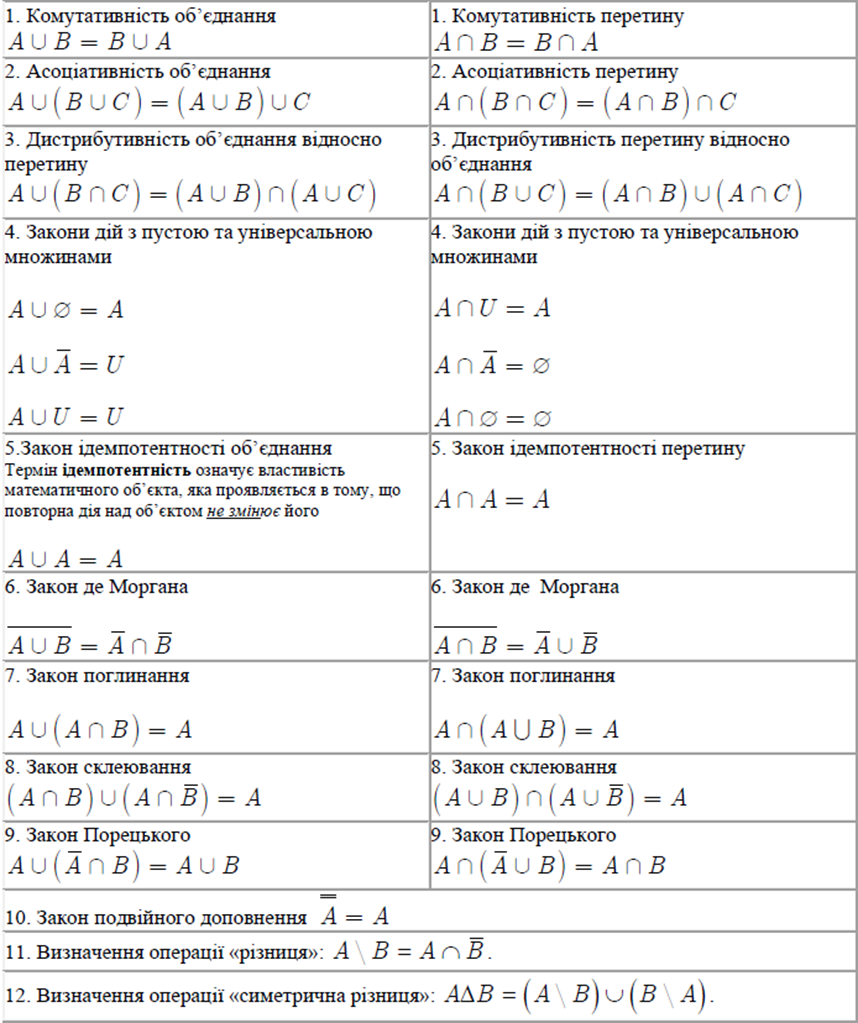
Об’єднання. Об'єднанням множин A і B називають множину, що складається із всіх тих елементів, які належать хоча б одній з множин A або B . Об'єднання множин A і B позначають A ∪ B . Це визначення рівносильне наступному: A ∪ B = {x x∈A або x∈B}.

Перетин. Перетином множин A і B називають множину, що складається із всіх тих елементів, які належать як множині A , так і множині B . Перетин множин A і B позначають A ∩ B . Це визначення рівносильне наступному:

A ∩ B = {x x∈A і x∈B}.

Доповнення. Доповненням (або абсолютним доповненням) множини A називають множину, що складається із всіх елементів універсальної множини, які не належать A . Доповнення множини A позначають ¬A. Це визначення рівносильне наступному: A =U - A = {x| x∈U і x A}.

Різниця. Різницею множин A і B (або відносним доповненням) називають множину, що складається із всіх елементів множини A , які не належать B .Різницю множин A і B позначають A - B або A\ B . Це визначення рівносильне наступному: A \ B { x| x∈ A и xB}.



**Спрощення логічного виразу**

D = (A ∩ B) ∪ (C ∩ B) ∪ (¬A ∩ ¬B) ∪ (¬B ∩ C)

Комутативність перетину:

(A ∩ B) ∪ (C ∩ B) ∪ (¬A ∩ ¬B) ∪ (¬B ∩ C) = (A ∩ B) ∪ (C ∩ B) ∪ (¬A ∩ ¬B) ∪ (C ∩ ¬B)

Закон склеювання:

(C ∩ B) ∪ (C ∩ ¬B) ∪ (A ∩ B) ∪ (¬A ∩ ¬B) = C ∪ (A ∩ B) ∪ (¬A ∩ ¬B)

За допомогою діаграми Венна-Ейлера визначив заперечення симетричної різниці:

C ∪ (A ∩ B) ∪ (¬A ∩ ¬B) = C ∪ ¬(A △ B)

**Блок-схеми, які відповідають алгоритмам, що використані в лабораторній роботі.**

Схемаоб'єднання

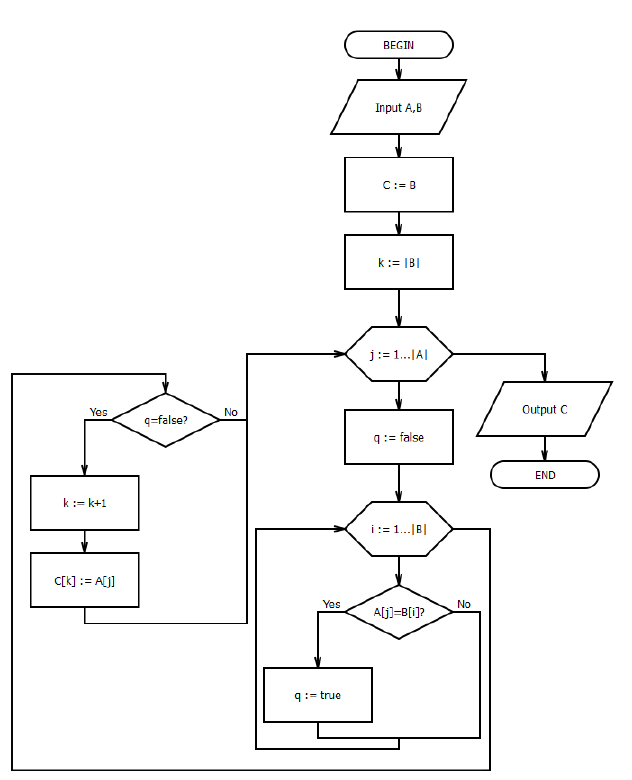
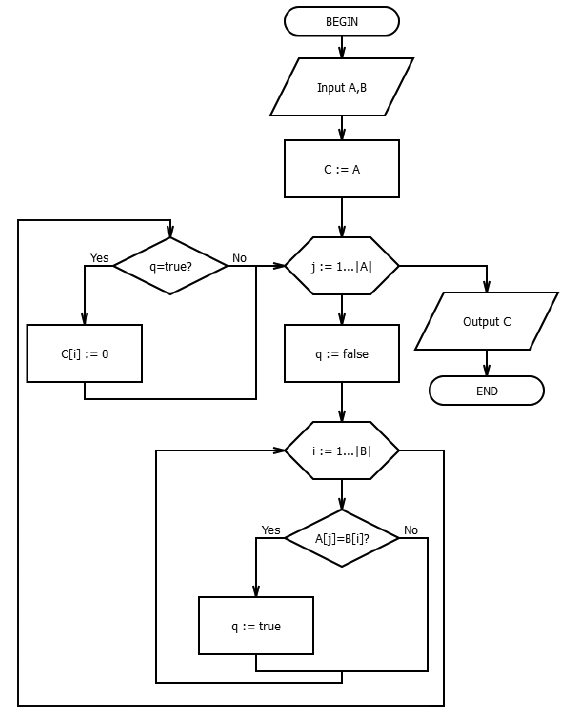


Схема різниці



**Фрагмент тексту програми, який написаний індивідуально**

Функція виводу покрокового виконання повного логічного виразу:

s1 = set(A & B)

s2 = set(C & B)

s3 = set((U - A) & (U -B))

s4 = set((U - B) & C)

s5 = set(s1 | s2)

s6 = set(s3 | s4)

s7 = set(s5 | s6)

def step():

global k

if (k == 0):

step\_txt.delete(1.0, END)

step\_txt.insert(INSERT, "Operation : (A ∩ B)\nResult:" + str(s1))

k += 1

elif (k == 1):

step\_txt.delete(1.0, END)

step\_txt.insert(INSERT, "Operation : (C ∩ B)\nResult:" + str(s2))

k += 1

elif (k == 2):

step\_txt.delete(1.0, END)

step\_txt.insert(INSERT, "Operation : (¬A ∩ ¬B)\nResult:" + str(s3))

k += 1

elif (k == 3):

step\_txt.delete(1.0, END)

step\_txt.insert(INSERT, "Operation : (¬B ∩ C)\nResult:" + str(s4))

k += 1

elif (k == 4):

step\_txt.delete(1.0, END)

step\_txt.insert(INSERT, "Operation : (A ∩ B) ∪ (C ∩ B)\nResult:" + str(s5))

k += 1

elif (k == 5):

step\_txt.delete(1.0, END)

step\_txt.insert(INSERT, "Operation : (¬A ∩ ¬B) ∪ (¬B ∩ C)\nResult:" + str(s6))

k += 1

elif (k == 6):

step\_txt.delete(1.0, END)

step\_txt.insert(INSERT, "Operation : D = (A ∩ B) ∪ (C ∩ B) ∪ (¬A ∩ ¬B) ∪ (¬B ∩ C)\nResult:" + str(s7))

k+=1

else:

k = 0

step\_txt.delete(1.0, END)

Функція покрокового виводу спрощеного логічного виразу:

s1 = set(U - (A.symmetric\_difference(B)))

s2 = set(C | s1)

def step():

global l

if (l == 0):

step\_txt.delete(1.0, END)

step\_txt.insert(INSERT, "Operation : ¬(A △ B)\nResult:" + str(s1))

l += 1

elif (l == 1):

step\_txt.delete(1.0, END)

step\_txt.insert(INSERT, "Operation : D = C ∪ ¬(A △ B)\nResult:" + str(s2))

l += 1

else:

l = 0

step\_txt.delete(1.0, END)

Написана мною функція виводу логічного виразу Z за допомогою циклів:

for i in U:

if i in A:

X.add(i)

for i in U:

if i not in B:

Y.add(i)

for i in U:

if (i in X) or (i in Y):

Z.add(i)

Функція виводу логічного виразу Z за допомогою вбудованих функцій Python:

def z2():

global A

global B

U = set([i for i in range(int(min\_en.get()), int(max\_en.get()) + 1)])

X = set(A)

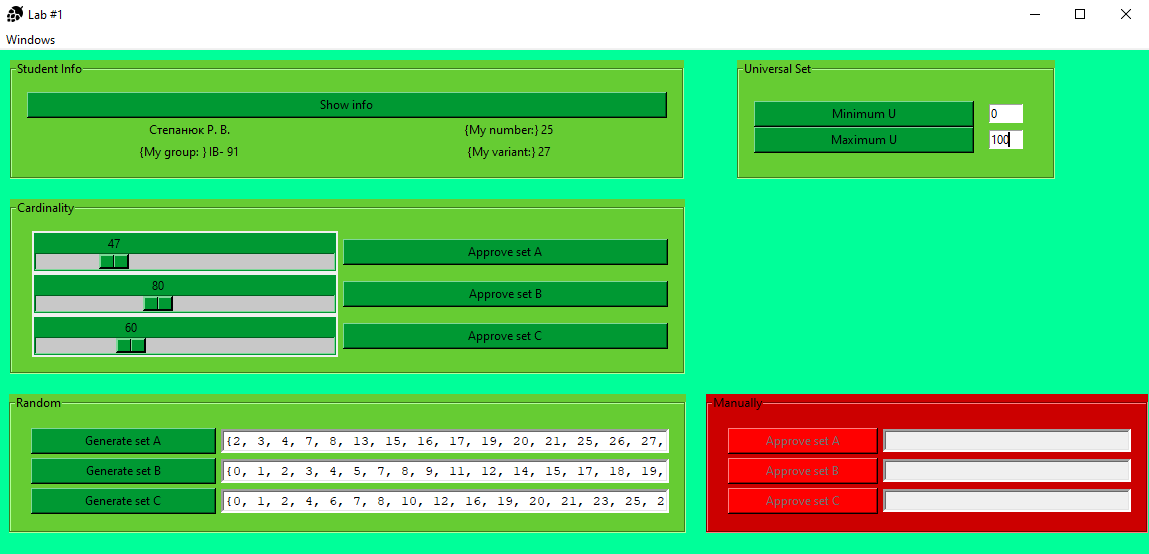
Y = set(U.difference(B))

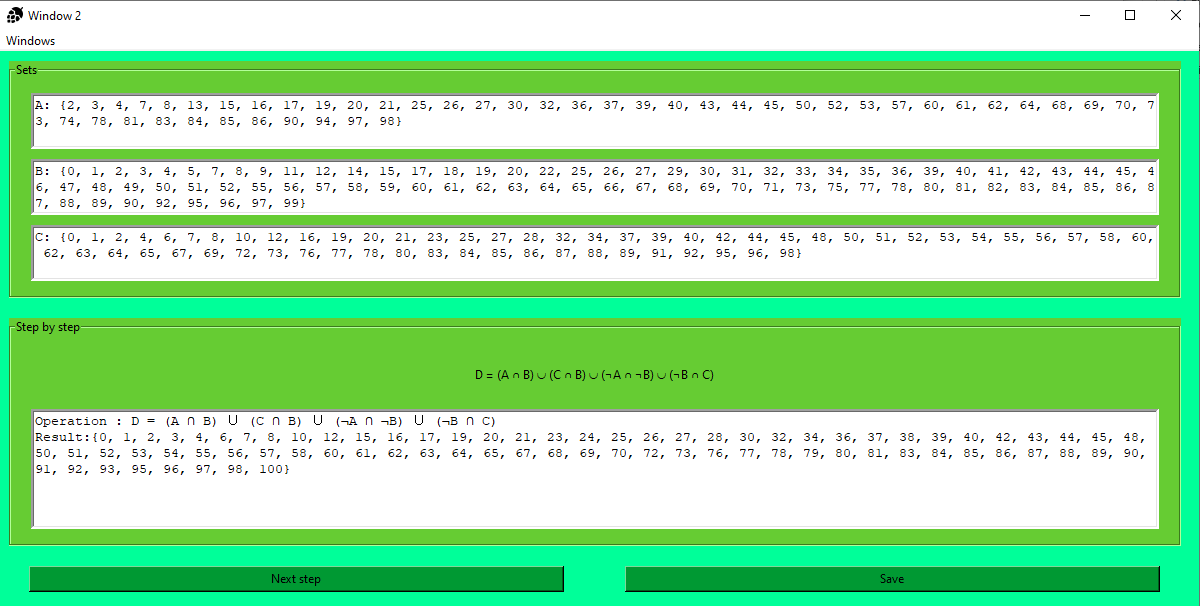
Z2 = set(X.union(Y))

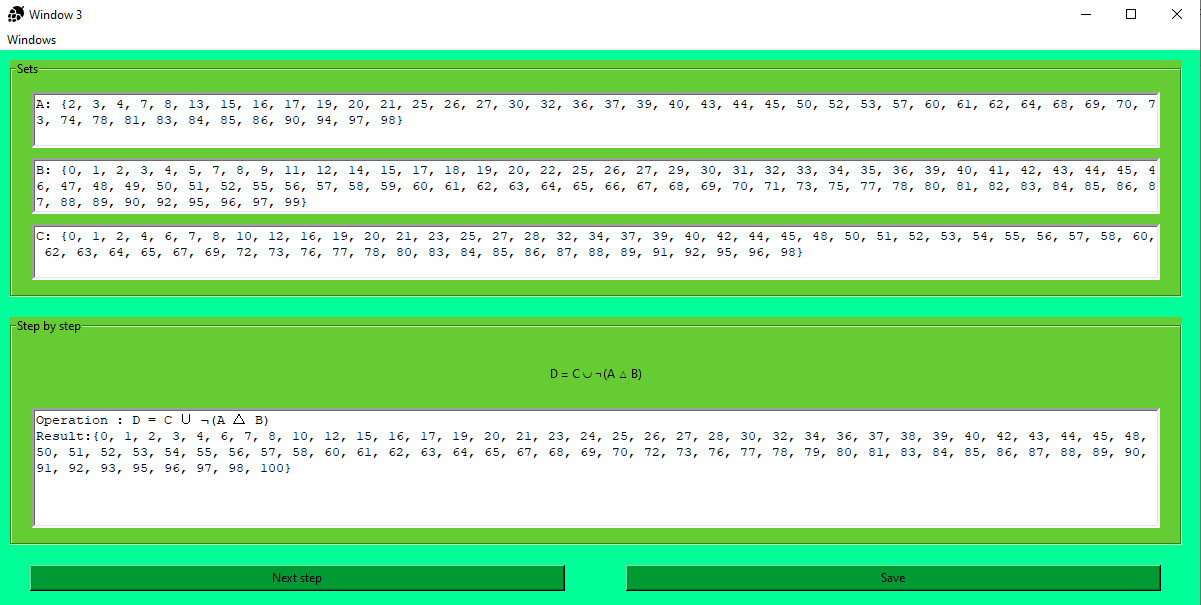
z2\_txt.delete(1.0, END)

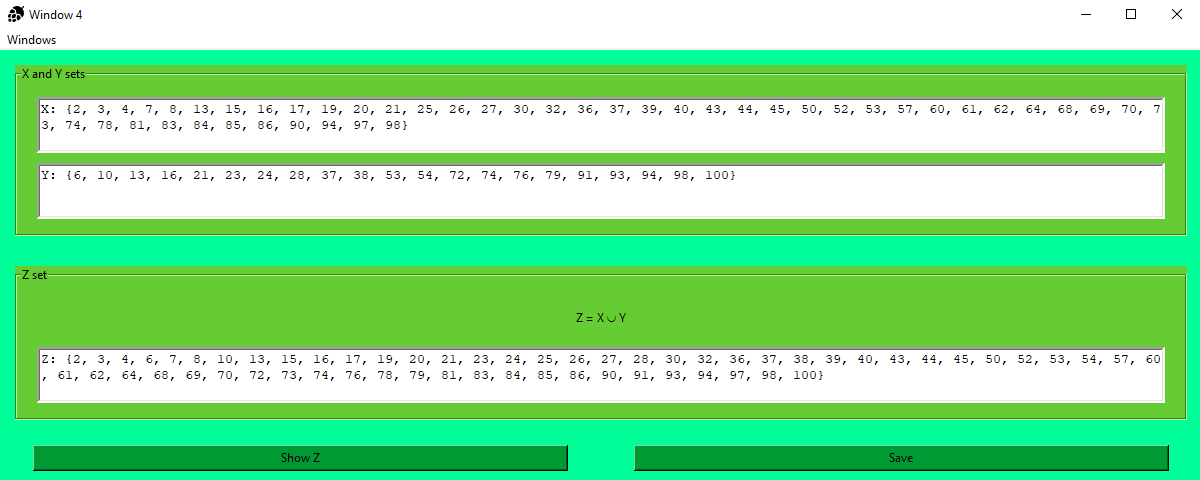
z2\_txt.insert(INSERT, Z2)

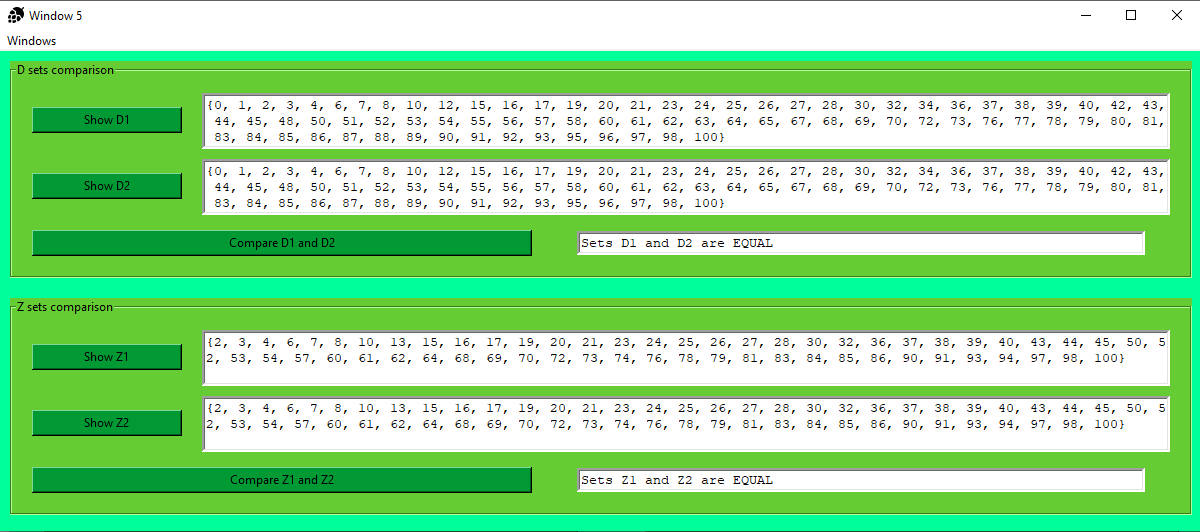
**Результати виконання програми з контрольним прикладом**











**Висновки**

Під час виконання лабораторної роботи серйозних проблем не виникло.

Єдина складність була в об’ємі роботи та дизайні інтерфейсу.