***UTN - TUP***

**Trabajo Práctico Integrador II**

**Matemática y Programación I**

**Estudiantes**

Lagos, Alejandro

Lahoz, Cristian Daniel

Maldonado, Ariana

Mubilla, Yanela

Ramallo, Gerónimo Gastón

**Comisión Nº8**

**Docente Titular**

Klimovsky, Ernesto

**Docente Tutor**

Comerci, Andrea

**JUNIO 2025**

## Objetivo

Profundizar la integración entre los contenidos de Matemática (conjuntos y lógica) y Programación 1 (estructuras condicionales, repetitivas y funciones), fortaleciendo también el trabajo en equipo, la comunicación clara y la responsabilidad individual en proyectos colaborativos.

## Parte 1 – Desarrollo Matemático (Conjuntos y Lógica)

### **Se detallaron los DNIs de cada uno de los miembros del equipo.**

1. Ramallo, Geronimo: 45413855
2. Mubilla, Yanela: 44011335
3. Lahoz, Cristian: 32084674
4. Lagos, Alejandro: 35569473
5. Maldonado, Ariana: 36184823

### **A partir de los DNIs, se formaron tantos conjuntos de dígitos únicos como miembros tiene el equipo.**

* 1. = {1,3,4,5,8}
  2. = {0,1,3,4,5}
  3. = {0,2,3,4,6,7,8}
  4. = {3,4,5,6,7,9}
  5. = {1,2,3,4,6,8}

### **Se realizaron entre esos conjuntos las operaciones de unión, intersección, diferencia (entre pares) y diferencia simétrica. Para cada una de estas operaciones, se dibujaron diagramas de Venn.**

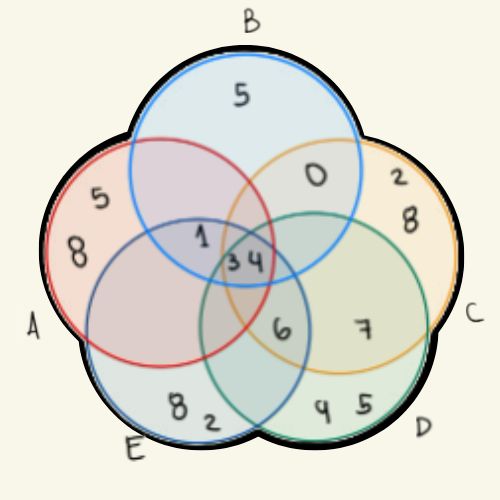
**Pasos**:

1. *Formación de Conjuntos de Dígitos Únicos*: los integrantes proporcionaron sus números de DNI, de los cuales se extrajeron todos los dígitos numéricos individuales, eliminando duplicados para formar conjuntos matemáticos puros.
2. *Unión de Conjuntos*: se realizó la operación de unión entre todos los conjuntos generados para identificar el universo completo de dígitos presentes en el grupo.
3. *Intersección de Conjuntos*: se calculó la intersección entre todos los conjuntos para identificar los dígitos comunes a todos los DNIs del grupo.
4. *Diferencia Simétrica*: se calculó la diferencia simétrica para identificar dígitos que son únicos a cada conjunto (presentes en uno pero no en todos).
5. *Diferencia*: se calculó la cantidad de combinaciones posibles entre pares de conjuntos con la fórmula: n\*(n-1) = 5\*(5-1) = 20 combinaciones. Se realizaron las 20 diferencias entre los pares de conjuntos.

### **UNIÓN**

**Definición:** La unión de dos conjuntos A y B es el conjunto que contiene todos los elementos que están en A, en B, o en ambos, sin repetir elementos.

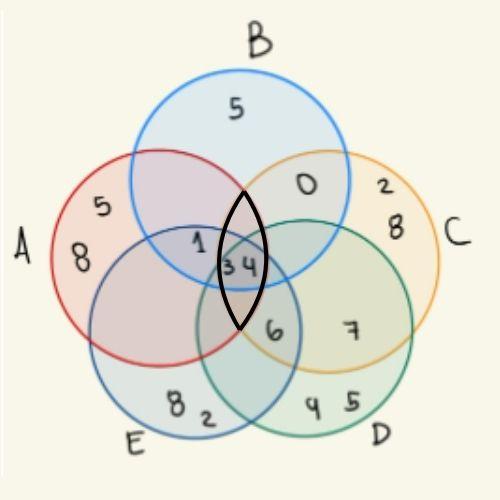
A U B U C U D U E = {0,1,2,3,4,5,6,7,8,9}



### **INTERSECCIÓN**

**Definición:** La intersección de dos conjuntos A y B es el conjunto formado por los elementos que pertenecen tanto a A como a B.

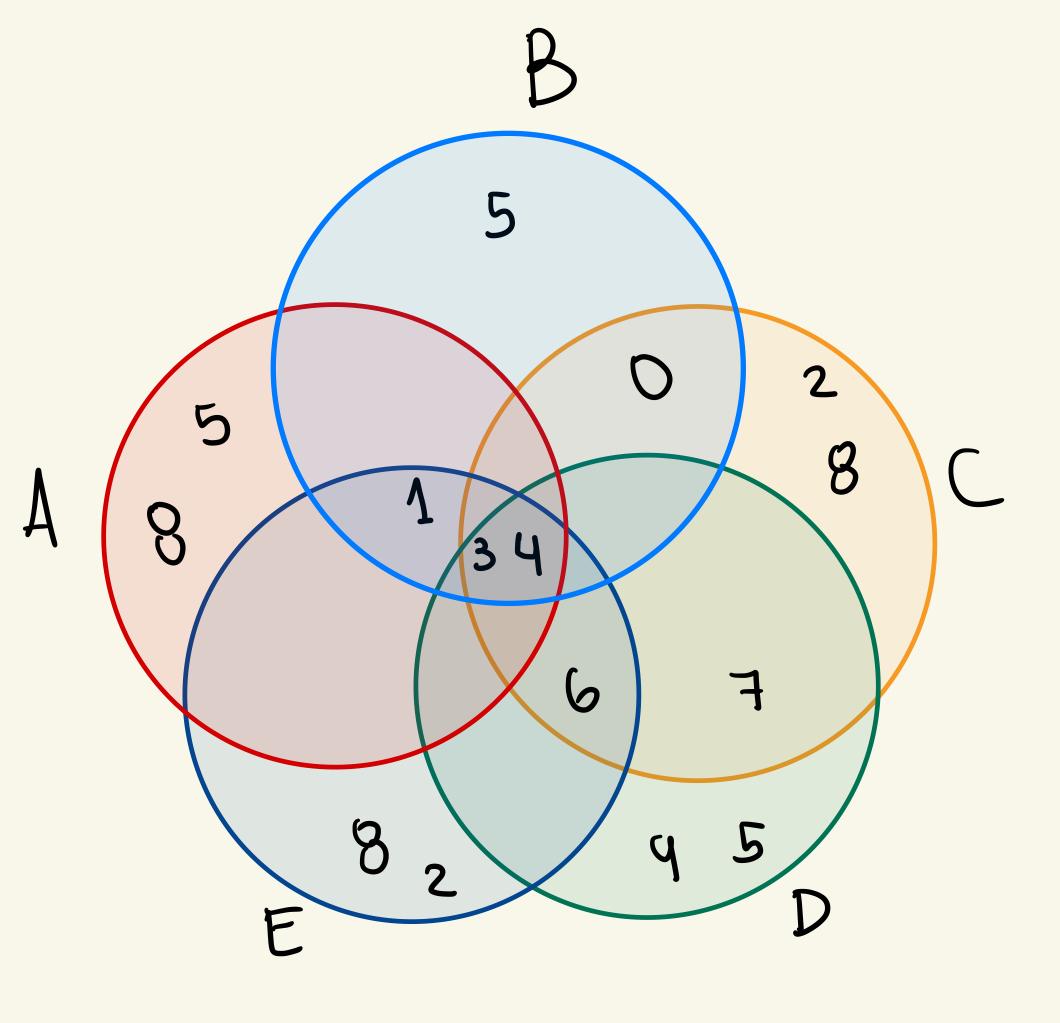
A ∩ B ∩ C ∩ D ∩ E = {3,4}



### **DIFERENCIA SIMÉTRICA**

**Definición:** La diferencia simétrica de dos conjuntos A y B es el conjunto de los elementos que pertenecen a A o a B, pero no a ambos al mismo tiempo.

A Δ B Δ C Δ D Δ E = {0,1,2,5,6,7,8,9}



1. **DIFERENCIA**

**Definición:** La diferencia de dos conjuntos A y B es el conjunto formado por todos los elementos que pertenecen a A, pero no a B.

|  |  |  |
| --- | --- | --- |
| **A - B** = {8} | **B - D** = {0,1} | **D - A** = {6,7,9} |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **A - C** = {1,5} | **B - E** = {0,5} | **D - B** = {6,7,9} |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **A - D** = {1,8} | **C - A** = {0,2,6,7} | **D - C** = {5,9} |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **A - E** = {5} | **C - B** = {0,2,6,7,8} | **D - E** = {5,7,9} |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **B - A** = {0} | **C - D** = {0,2,8} | **E - A** = {2,6} |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **B - C** = {0,1,5} | **C - E** = {0,7} | **E - B** = {2,6,8} |
|  |  |  |

|  |  |
| --- | --- |
| **E - C** = {1} | **E - D** = {1,2,8} |
|  |  |

### **Se desarrollaron dos expresiones lógicas en lenguaje natural.**

* Si todos los conjuntos contienen al menos n dígitos en común, entonces el grupo tiene n dígitos comunes.
* Si hay más conjuntos con cantidad impar de elementos que conjuntos con cantidad par, entonces el grupo se etiqueta como “grupo impar”.

## Parte 2 – Desarrollo del Programa en Python

El programa debe implementar varias de las ideas trabajadas en papel. Debe incluir:

1. **Operaciones con DNIs**

* Ingreso de los DNIs (reales o ficticios).
* Generación automática de los conjuntos de dígitos únicos.
* Cálculo y visualización de: unión, intersección, diferencias y diferencia simétrica.
* Conteo de frecuencia de cada dígito en cada DNI utilizando estructuras repetitivas.
* Suma total de los dígitos de cada DNI.
* Evaluación de condiciones lógicas (condicionales), vinculadas con las expresiones escritas.

**Ejemplos:**

* Si un dígito aparece en todos los conjuntos, mostrar "Dígito compartido".
* Si algún conjunto tiene más de 6 elementos, mostrar "Diversidad numérica alta".

**B. Operaciones con años de nacimiento**

* Ingreso de los años de nacimiento (Si dos o mas integrantes del grupo tienen el mismo año, ingresar algún dato ficticio, según el caso).
* Contar cuántos nacieron en años pares e impares utilizando estructuras repetitivas.
* Si todos nacieron después del 2000, mostrar "Grupo Z".
* Si alguno nació en año bisiesto, mostrar "Tenemos un año especial".
* Implementar una función para determinar si un año es bisiesto.
* Calcular el producto cartesiano entre el conjunto de años y el conjunto de edades actuales.

**Parte 3 – Video de Presentación**

Duración estimada entre 5 y 10 minutos. Todos los integrantes deben presentarse en cámara, mostrar el programa funcionando y explicar la parte que realizaron. También deben comentar brevemente qué aprendieron al combinar matemática y programación.

**Entrega final**

1. Archivo PDF con: desarrollo de conjuntos y operaciones, todos los diagramas de Venn, expresiones lógicas redactadas, y tareas de cada integrante explicadas por escrito.

2. Archivo con extensión .py que contenga el programa en Python.

3. Video grupal subido en lo posible a YouTube.

4. Documento adicional con los nombres de los integrantes, descripción de lo que hizo cada uno y la relación entre las expresiones lógicas escritas y el código implementado.