

Trabajo Integrador 2: Matemática y Programación

Alumno: Máximo Ponce – maximoponce31@gmail.com

Alumno: Lucio Rivas – lucior0900@gmail.com

Materia: Matemática y Programación

Fecha de Entrega: 13/06/2025

Comisión: 4

Índice

1. Introducción
2. Conjuntos y Operaciones
3. Expresiones Lógicas
4. Diagramas de Venn
5. Consideraciones Finales
6. Anexos

1. Introducción

Este trabajo integrador tiene como objetivo relacionar contenidos fundamentales de matemática —como conjuntos y lógica— con estructuras básicas de programación en Python. Partimos desde los números de DNI de los integrantes para construir conjuntos de dígitos únicos, sobre los cuales se realizan operaciones matemáticas, análisis lógicos y su posterior implementación en un programa.

Además, se incorporan aspectos de programación estructurada como condicionales, bucles y funciones, incluyendo el análisis de años de nacimiento de los integrantes. El objetivo general es lograr una comprensión práctica de cómo la lógica matemática se transforma en instrucciones de código.

2. Conjuntos y Operaciones

A partir de los DNIs de los integrantes del grupo, se obtuvieron los siguientes conjuntos de dígitos únicos:

- **Alumno 1 – DNI 43322920** → Conjunto A: {0, 2, 3, 4, 9}
- **Alumno 2 – DNI 46351171** → Conjunto B: {1, 3, 4, 5, 6, 7}

Operaciones realizadas

- **Unión ($A \cup B$):** {0, 1, 2, 3, 4, 5, 6, 7, 9}
- **Intersección ($A \cap B$):** {3, 4}
- **Diferencia ($A - B$):** {0, 2, 9}
- **Diferencia ($B - A$):** {1, 5, 6, 7}
- **Diferencia Simétrica ($A \Delta B$):** {0, 1, 2, 5, 6, 7, 9}

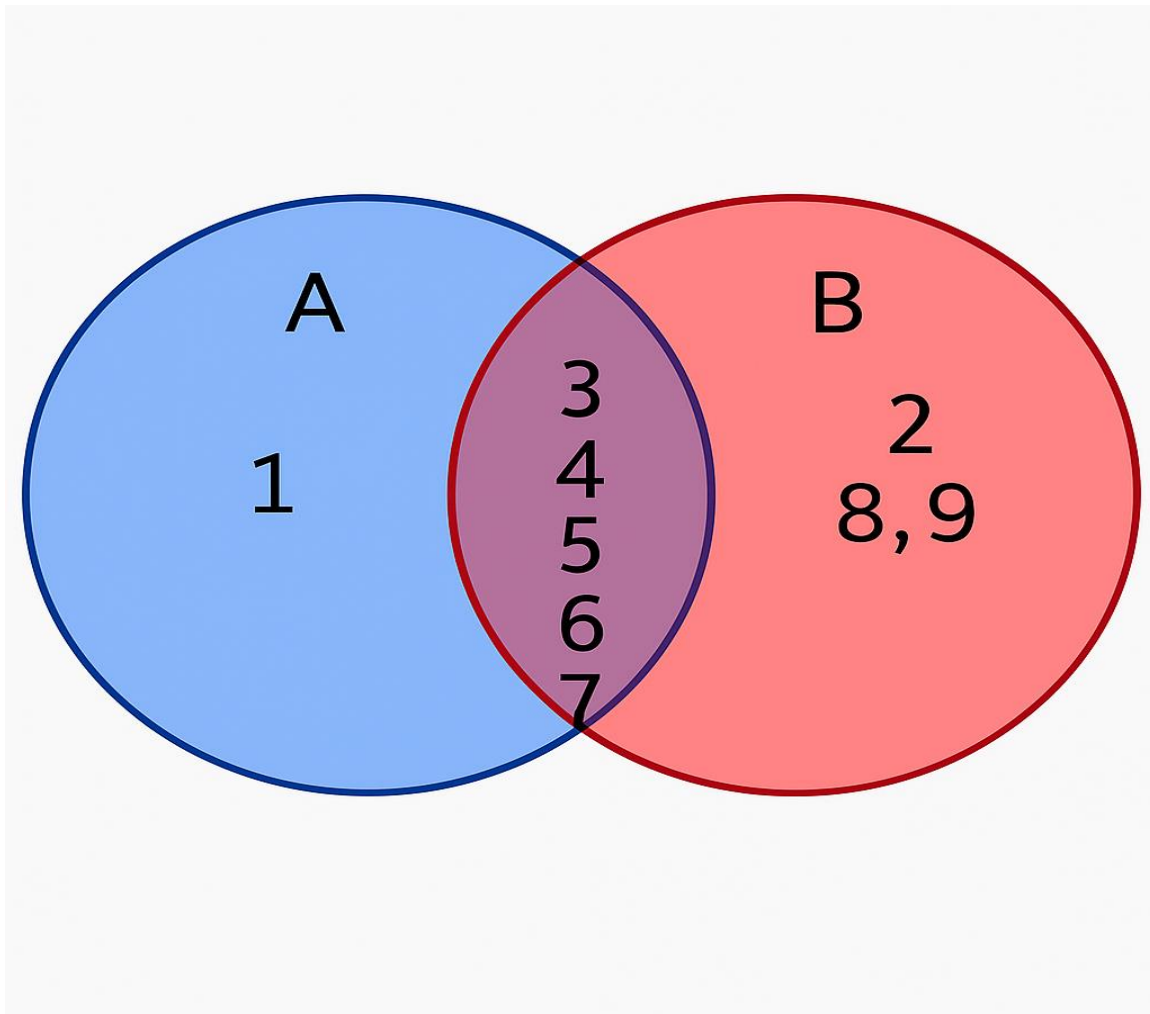
Estas operaciones fueron implementadas tanto en forma teórica como en el código Python.

3. Expresiones Lógicas

A partir de los conjuntos anteriores, se definieron expresiones lógicas en lenguaje natural que luego fueron aplicadas y verificadas en el código Python.

1. **"Si todos los conjuntos tienen al menos 5 elementos, entonces se considera que hay una alta diversidad numérica."**
→ Conjunto A tiene 5 elementos. Conjunto B tiene 6. Se cumple la condición.
2. **"Si algún dígito aparece en todos los conjuntos, se marca como dígito común."**
→ Los dígitos 3 y 4 están presentes en ambos conjuntos. Se cumple la condición.
3. **"Si algún conjunto contiene el dígito 0, se considera un grupo con ceros."**
→ Conjunto A contiene el dígito 0. Se cumple la condición.
4. **"Si la intersección de todos los conjuntos tiene exactamente dos elementos, se considera coincidencia limitada."**
→ La intersección tiene exactamente 2 elementos (3 y 4). Se cumple la condición.

4. Diagrama de Venn



Leyenda:

- Zona A (azul): {0, 2, 3, 4, 9}
- Zona B (roja): {1, 3, 4, 5, 6, 7}
- Intersección: {3, 4}

5. Consideraciones Finales

Este trabajo integrador nos permitió observar cómo conceptos matemáticos pueden llevarse a la práctica con programación. Mediante operaciones de conjuntos, condiciones lógicas y estructuras como funciones y bucles, se construyó un programa capaz de analizar automáticamente datos reales (como los DNIs y años de nacimiento).

Al trabajar en equipo, cada integrante asumió tareas específicas:

- Máximo se encargó del desarrollo teórico, operaciones y expresiones lógicas, así como de la redacción del documento.
- Lucio se encargó del desarrollo del programa en Python, aplicando las condiciones descritas y explicándolas en el video.

6. Anexos

Código fuente (.py)

El archivo(analisis_dni_anios.py) contiene todo el desarrollo programado en Python. Entre sus funciones se incluyen:

- Obtención de DNIs y conversión a conjuntos.
- Cálculo de operaciones entre conjuntos.
- Evaluación de condiciones lógicas como presencia del dígito 0 o intersecciones.
- Frecuencia de aparición de dígitos y suma total por DNI.
- Análisis de años de nacimiento: par/impar, grupo Z, año bisiesto.
- Cálculo del producto cartesiano entre años y edades actuales.

El código completo se incluye como archivo adjunto en la entrega final.

Video de Presentación