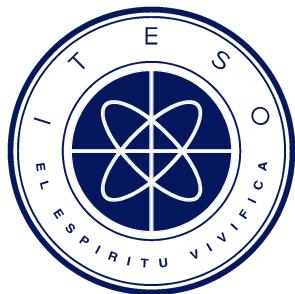


INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE OCCIDENTE

FUNDAMENTOS DE CIENCIAS COMPUTACIONALES



CALCULADORA DE CONJUNTOS

Presentan:

Ana Carolina Arellano Valdez

Luis Raúl Acosta Mendoza

Arlyn Linette Medina García

Profesora:

Silvia Piña Romero

Fecha:

Proyecto

Relevancia de los conjuntos en las Ciencias Computacionales

Para comenzar, los conjuntos son una agrupación definida de elementos. Estos pueden llegar a ser fundamentales en la computación, ya que estos provienen de principios matemáticos a la vez que agrupan varias ramas de las matemáticas. A pesar de que el concepto de los conjuntos puede llegar a parecer sencillo, estos conforman la base de las matemáticas actuales en donde llegan a contribuir para la construcción de números. Dentro de las ciencias computacionales, los conjuntos son fundamentales para varios procesos como lo son los circuitos computacionales, la programación lógica a la vez que en la optimización y en el análisis de los programas. Ya que todo sistema computacional, sin importar el nivel de complejidad, está compuesto por circuitos electrónicos basados en el lenguaje binario, los conjuntos llegan a tener una gran aportación.

Explicación técnica de la programación.

- Nuestro programa principalmente esta conformado por varias funciones, cada una relacionada a las diferentes operaciones que se pueden hacer con los conjuntos.

Función Intersección:

- En esta función primeramente se establece una lista llamada “intersection” la cual va a ser el resultado que se le va a devolver al usuario. Después se entra en un ciclo for el cual va recorriendo cada valor del conjunto A. En el mismo ciclo se encuentra una condicional if que con cada valor del conjunto A, revisa si se encuentra en el conjunto B. Al esto ser verdadero, el valor se agrega a la lista “intersection” con el comando append así resultando con los valores que ambos conjuntos comparten.

Función Unión:

- En esta función de igual manera, se crea una lista llamada unión la cual va a ser el resultado que se le va a devolver al usuario. El programa consiste en dos ciclos for en donde el primero une cada valor del conjunto A en la lista unión. El segundo ciclo revisa cada valor del conjunto de B y entra en una condicional en donde si no hay un valor en la lista unión, se le agrega. Así se resulta con una lista en donde se encuentran todos los valores de los conjuntos A y B sin repeticiones.

Función Diferencia A-B y Función Diferencia B-A:

- En esta función también se crea una lista el cual será el resultado que se le devolverá al usuario llamado difference. Este programa consiste en un ciclo for en donde se revisa cada valor de un conjunto, depende del A o del B de la opción que el usuario haya escogido. Se entra en una condicional if en la que, si el valor de ese conjunto no se encuentra en el otro conjunto, se le agrega a la lista. Con esto se resulta una lista con los valores que están solamente en el conjunto A, si es que el usuario escogió la opción A-B, o con los valores que solamente se encuentran en el conjunto B, si es que el usuario escogió la opción B-A.

Función Diferencia Simétrica:

- En esta función se llama a otra función llamada difference2 en la que se hace lo mismo que la función de diferencia de A-B Y B-A con la diferencia que en este se hace ambos

a la misma vez. Esto para que se acomodarán mejor los valores en la lista y que no hubiera ningún inconveniente con los corchetes.

Función Complemento de A y Función Complemento B:

- Esta función consiste en un ciclo for en el que se revisa cada valor del conjunto A o B, dependiendo de lo que el usuario haya escogido. Se entra en una condicional if en donde si el valor de ese no se encuentra en el otro o en el universo que va hasta 100, se agrega a la lista complement . Con esto se resulta con una lista que no tiene ningún valor del conjunto A, si es que el usuario escogió complemento de A, o con una lista que no tiene ningún valor del conjunto B, si es que el usuario escogió la opción de complemento de B.

Función Producto Cartesiano AxB, BxA, AxA, BxB:

- Esta función principalmente consiste en dos ciclos for de los cuales el primero revisa valor por valor el primer conjunto mientras que el segundo ciclo el segundo conjunto. Cada vez que se entra a estos dos ciclos, se va agregando cada valor del primer conjunto con cada valor del segundo conjunto. Esto nos resulta con todas las combinaciones posibles del producto cartesiano de los conjuntos que el usuario haya escogido.

Función Conjunto Potencia A y Función Conjunto Potencia B:

- En esta función lo que primero se checa es que el conjunto que el usuario haya escogido no este vacío, si lo está, simplemente regresara un vacío. Por otro lado, si esto no es verdadero, lo que se hace regresan los valores del conjunto potencia a través de una operación recursiva. En esta operación recursiva se toma en cuenta todas las combinaciones posibles que se puede tener en el conjunto que el usuario escogió.

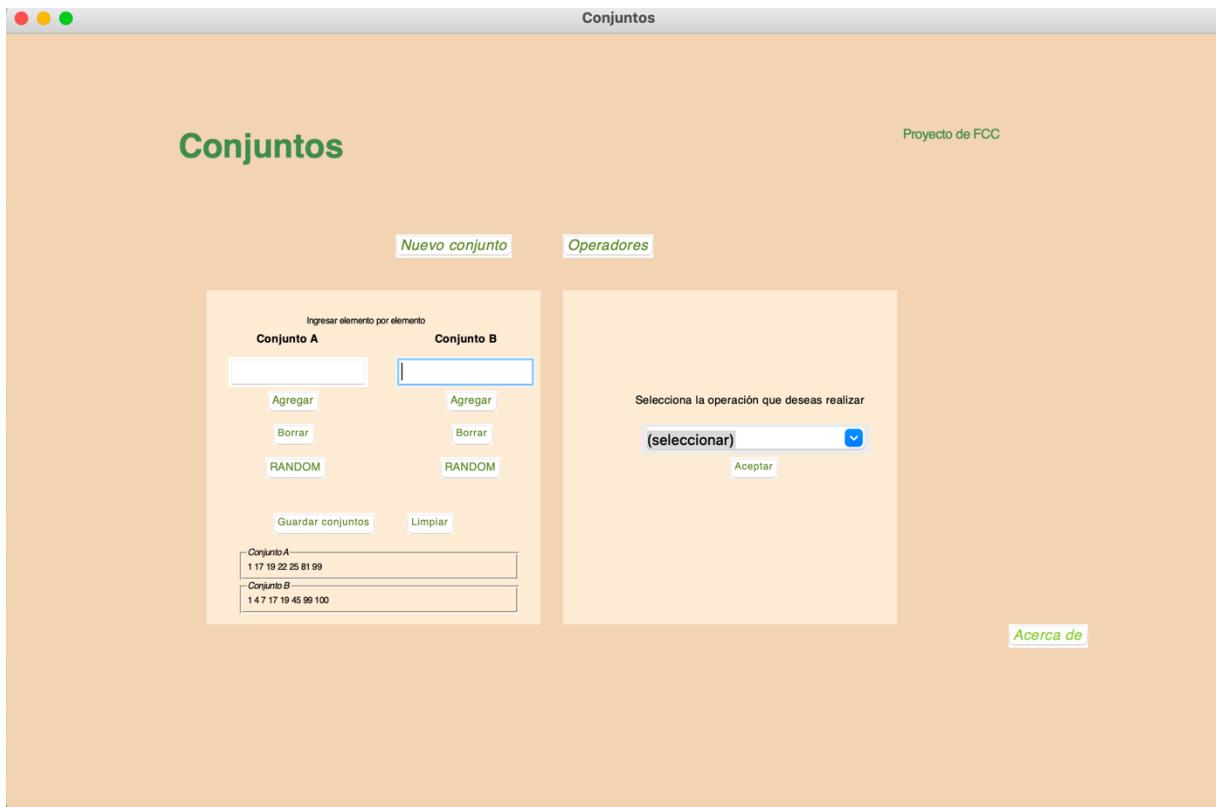
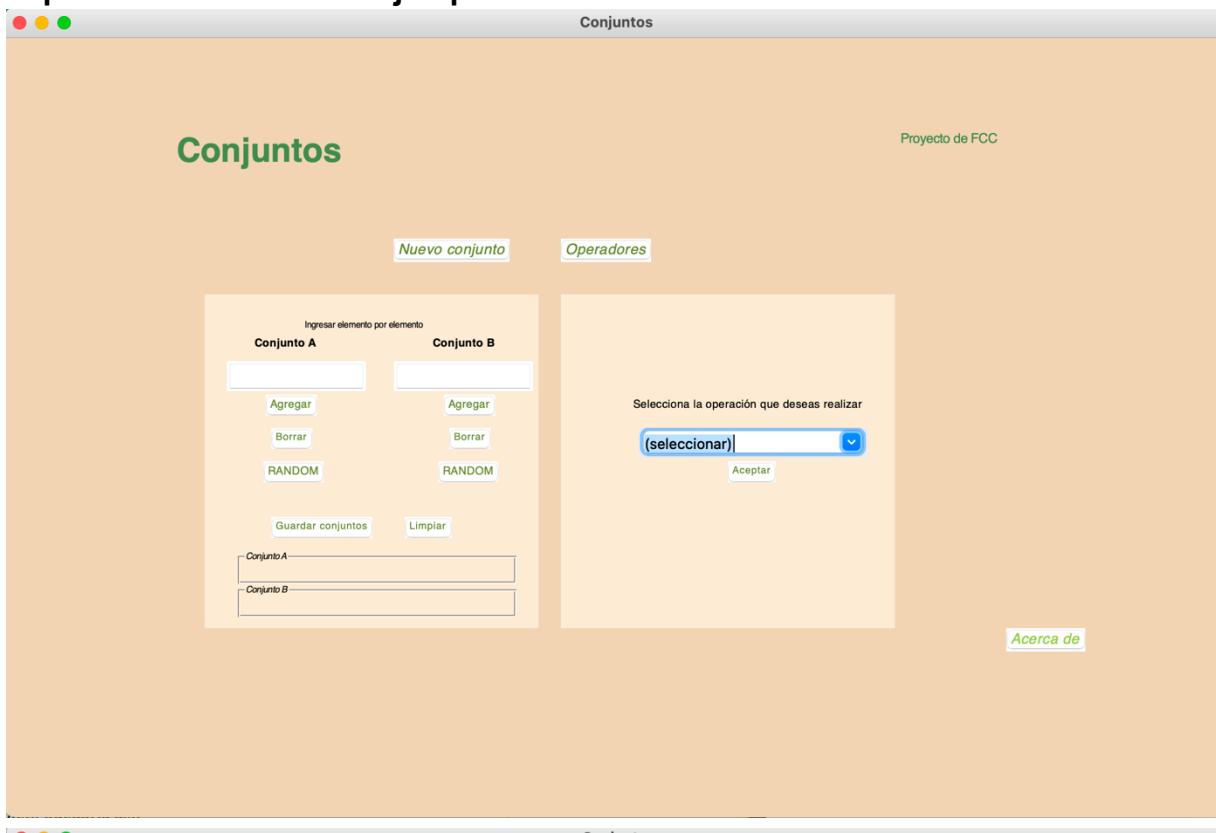
Función Cardinalidad A y Función Cardinalidad B:

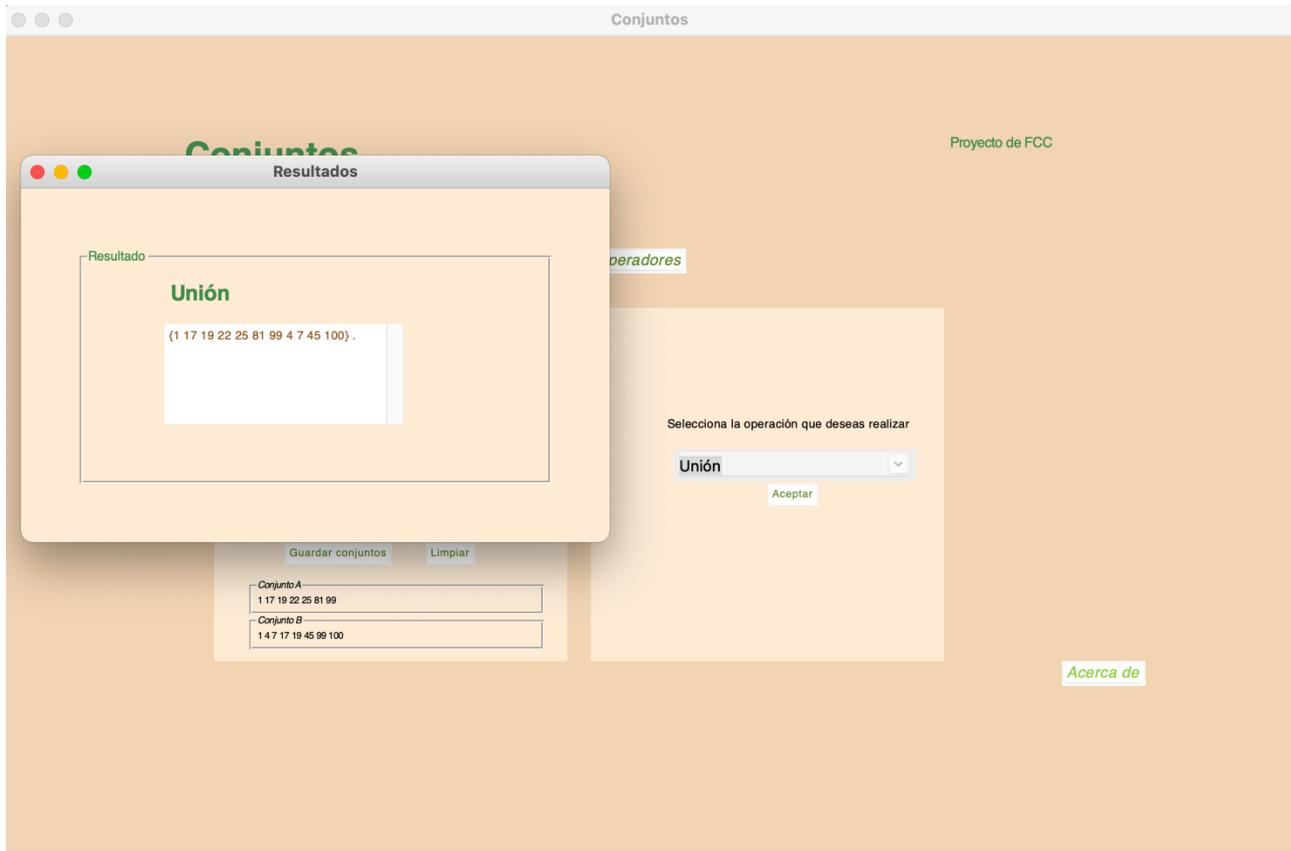
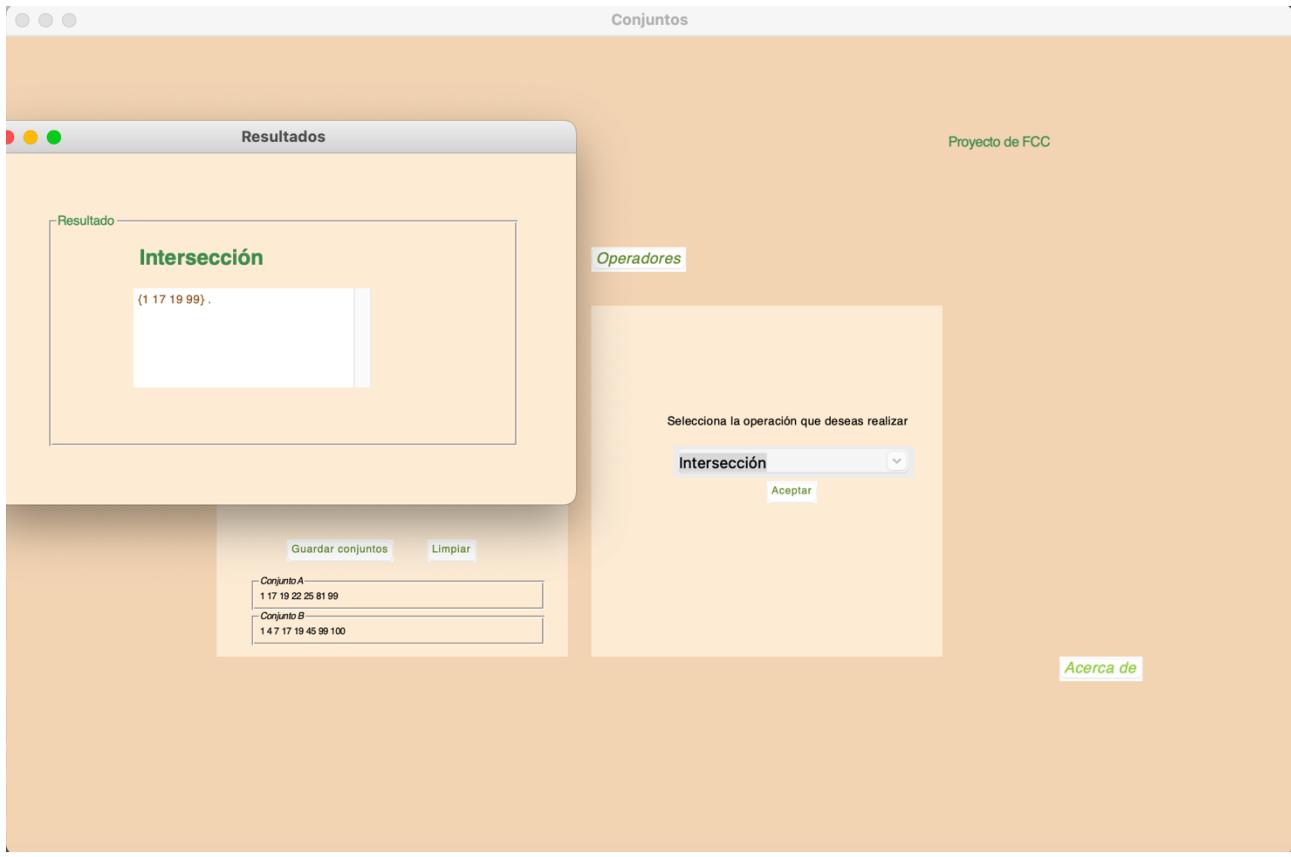
- Esta función nada más consiste en una línea de código en la que básicamente se devuelve el valor de la longitud del que el usuario haya escogido con el comando len.

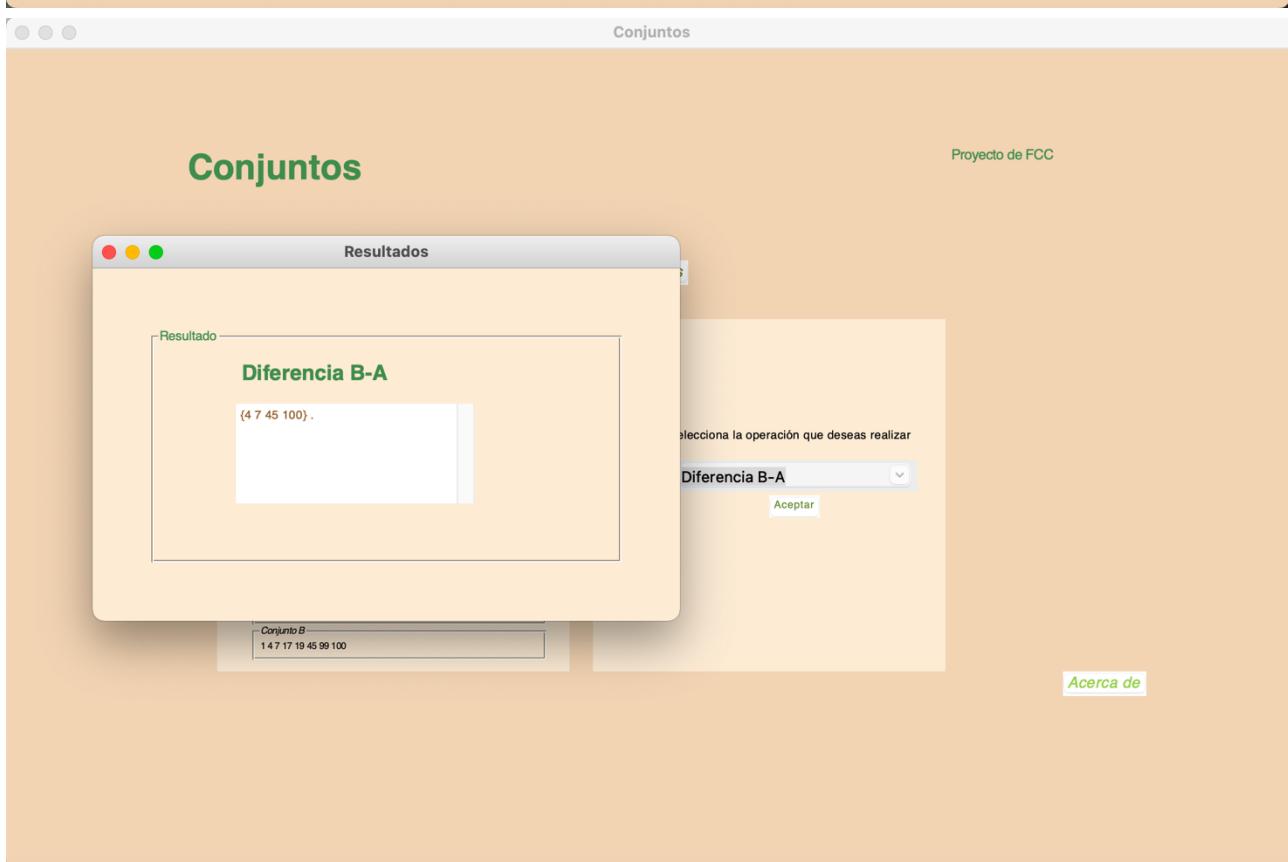
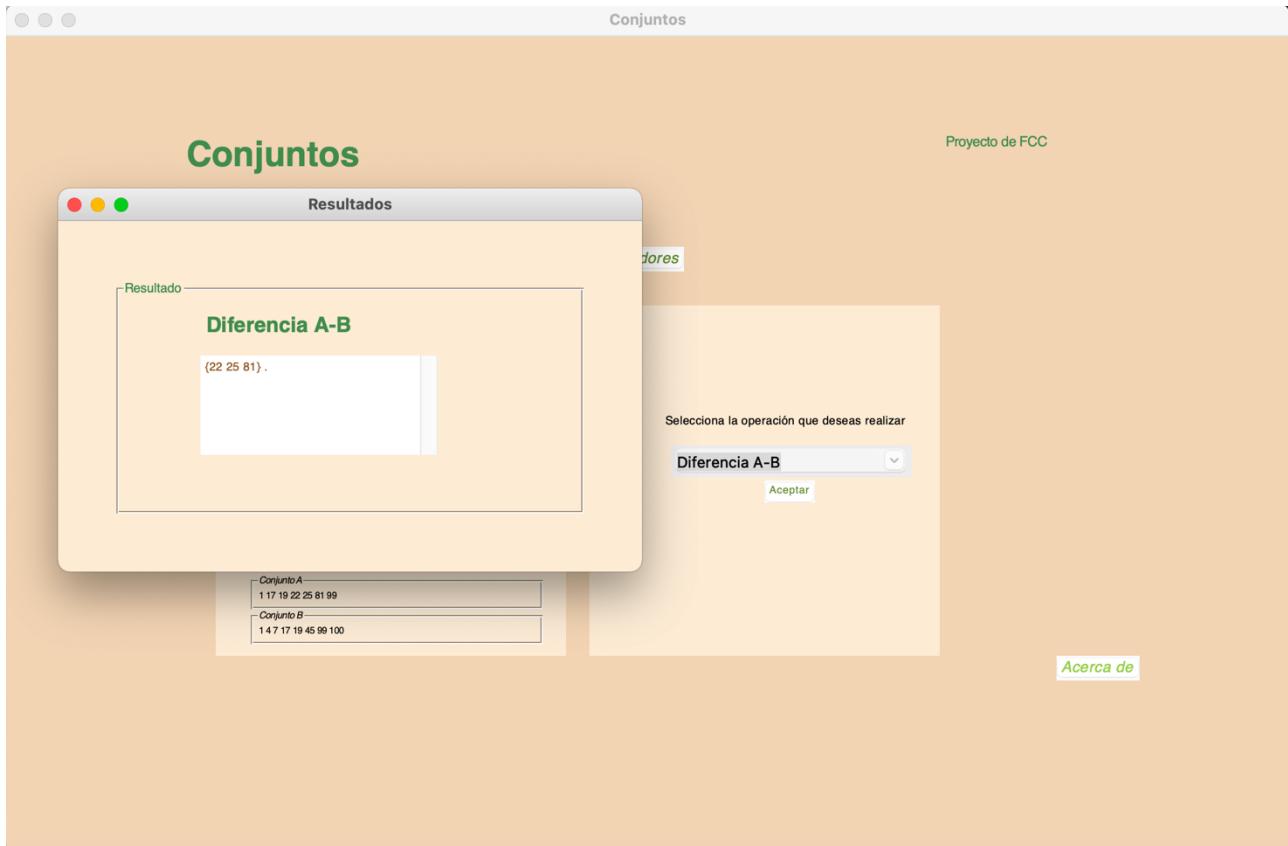
Función Contención a $A \subseteq B$ y Función Contención a $B \subseteq A$:

- En esta función lo que se hace es checar valor por valor del conjunto que se está revisando si está contenido en el otro con un ciclo for. Si todos sus valores se encuentran en el otro conjunto, se le da a una variable t el valor de verdadero y si no simplemente se regresa el valor falso. Después se entra en un if en donde si la longitud de ambos conjuntos es igual, significa que son conjuntos propios por lo que regresaría un falso, mientras que, si no los son, regresaría un verdadero.

Capturas de Pantalla de Ejemplo:







Conjuntos

Proyecto de FCC

Resultados

Operadores

Selección la operación que deseas realizar

Complemento de A

Aceptar

Guarda conjuntos Limpiar

Conjunto A
1 17 19 22 25 81 99

Conjunto B
1 4 7 17 19 45 99 100

Acerca de

Conjuntos

Proyecto de FCC

Resultados

Operadores

Selección la operación que deseas realizar

Complemento de B

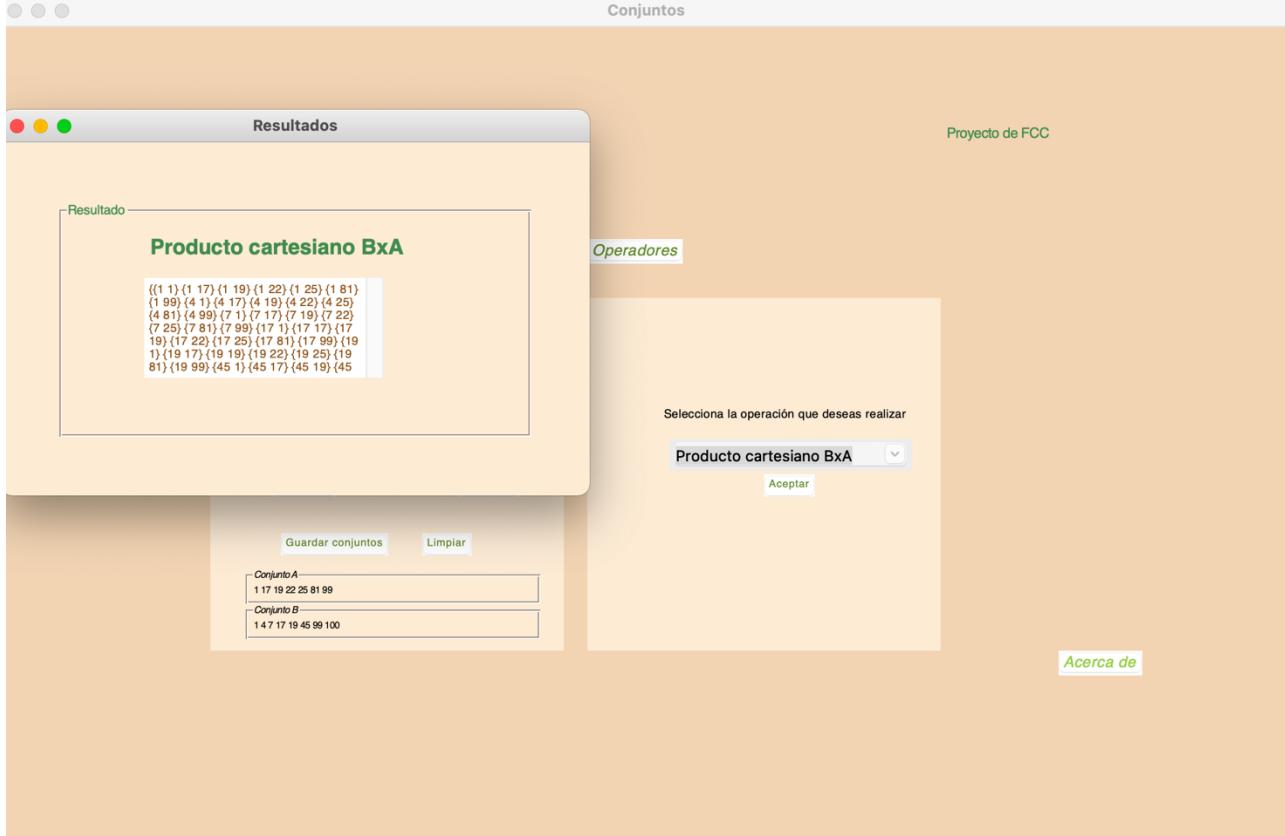
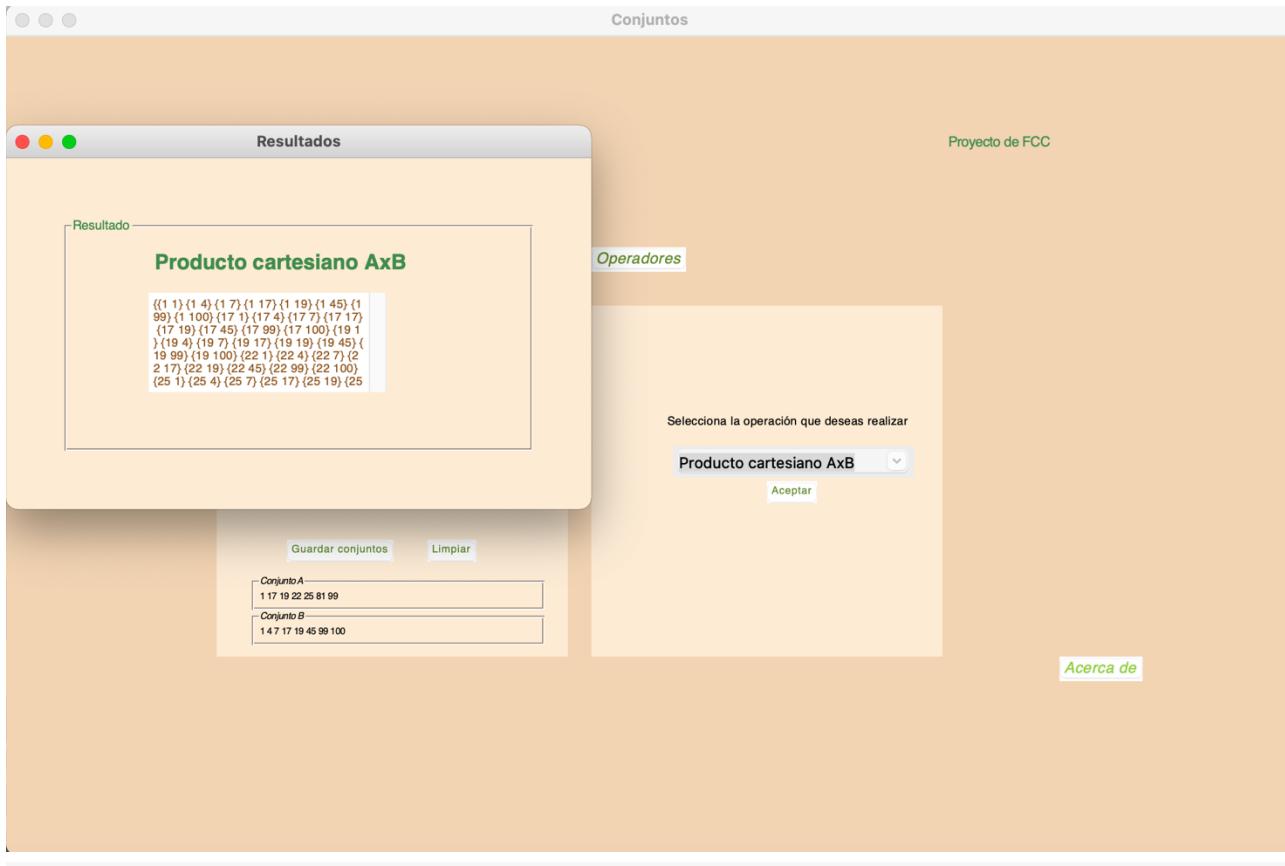
Aceptar

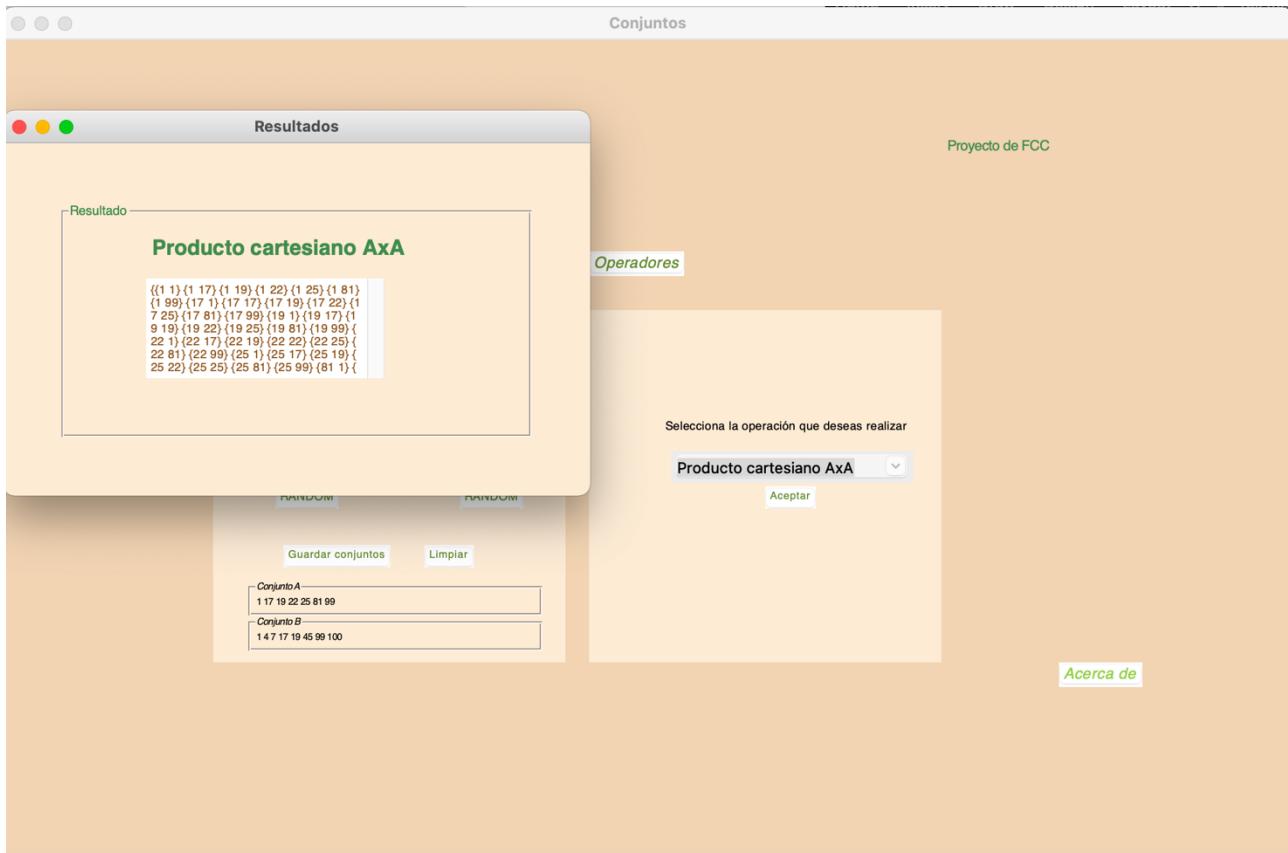
Guarda conjuntos Limpiar

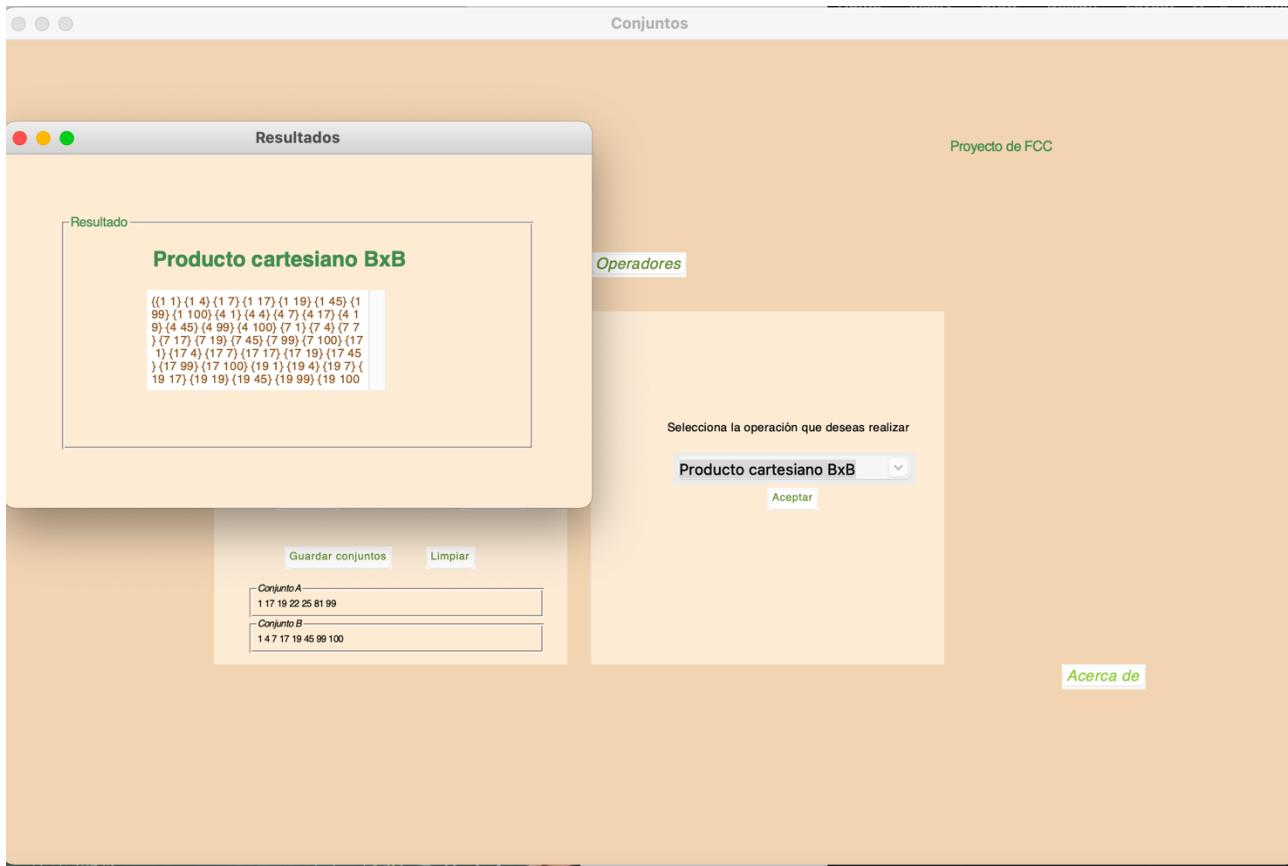
Conjunto A
1 17 19 22 25 81 99

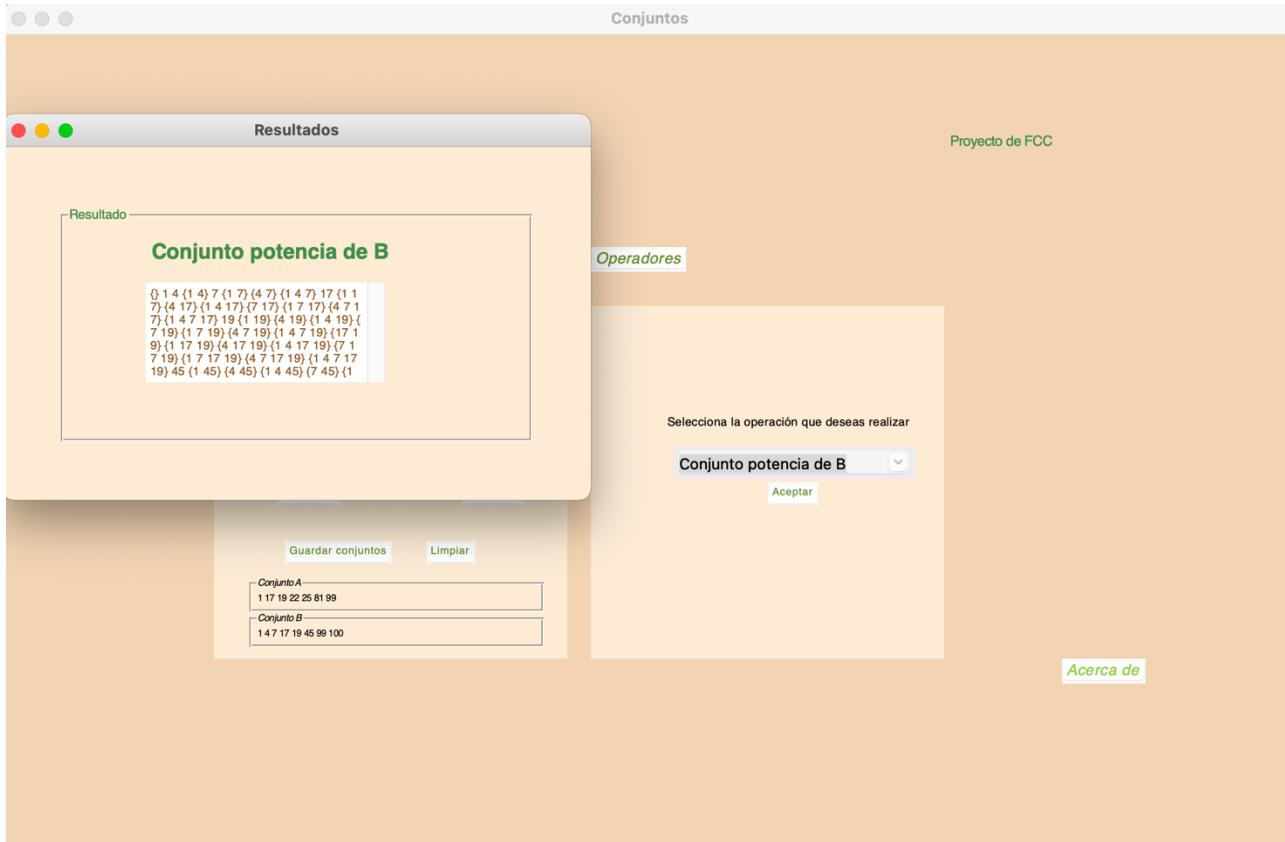
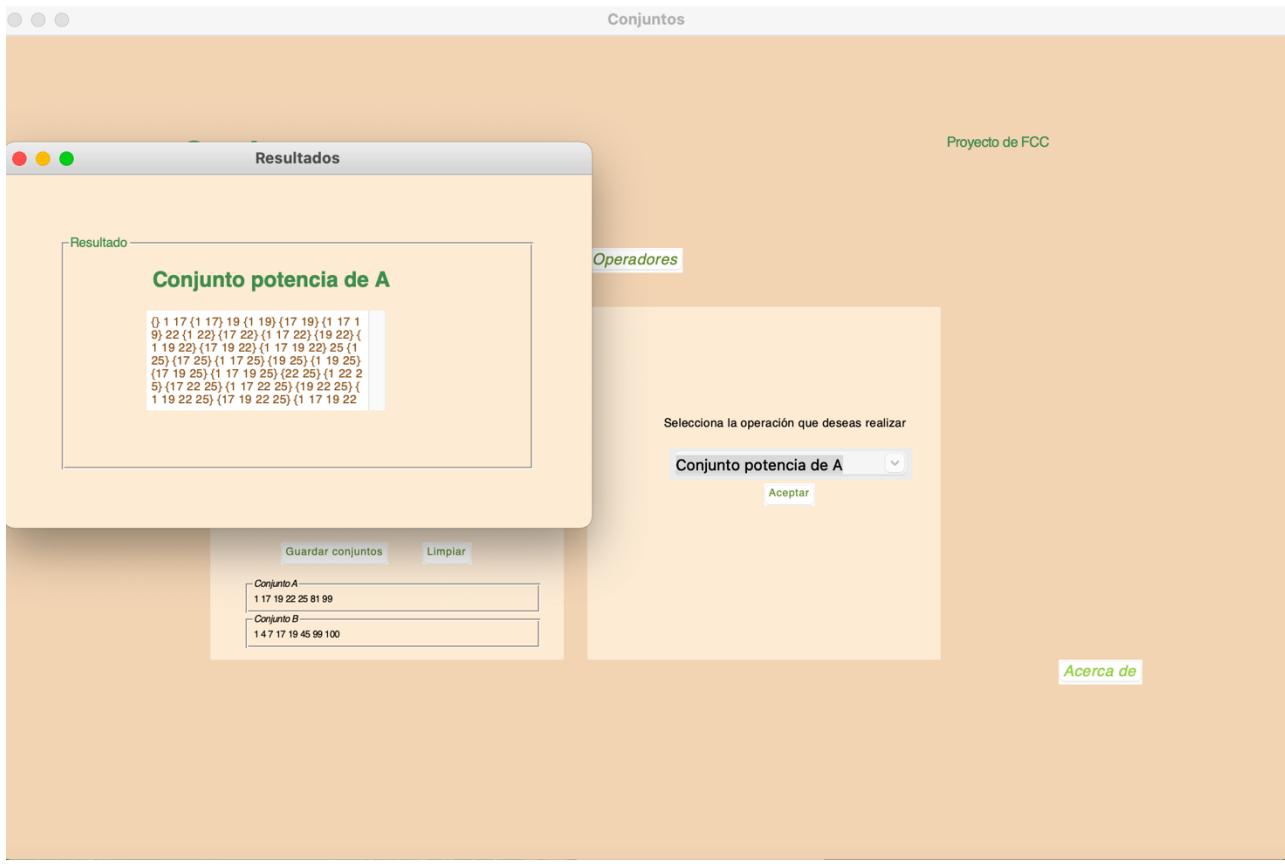
Conjunto B
1 4 7 17 19 45 99 100

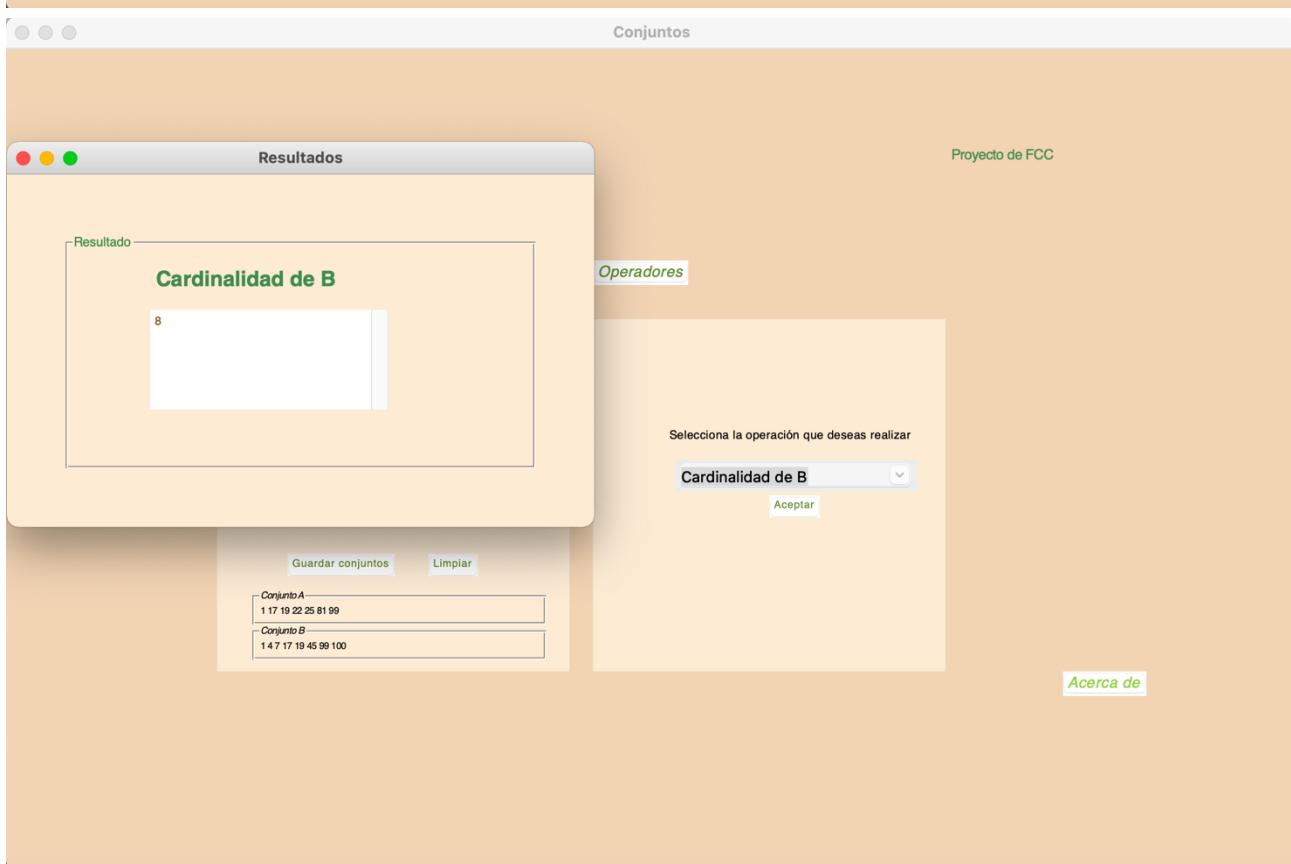
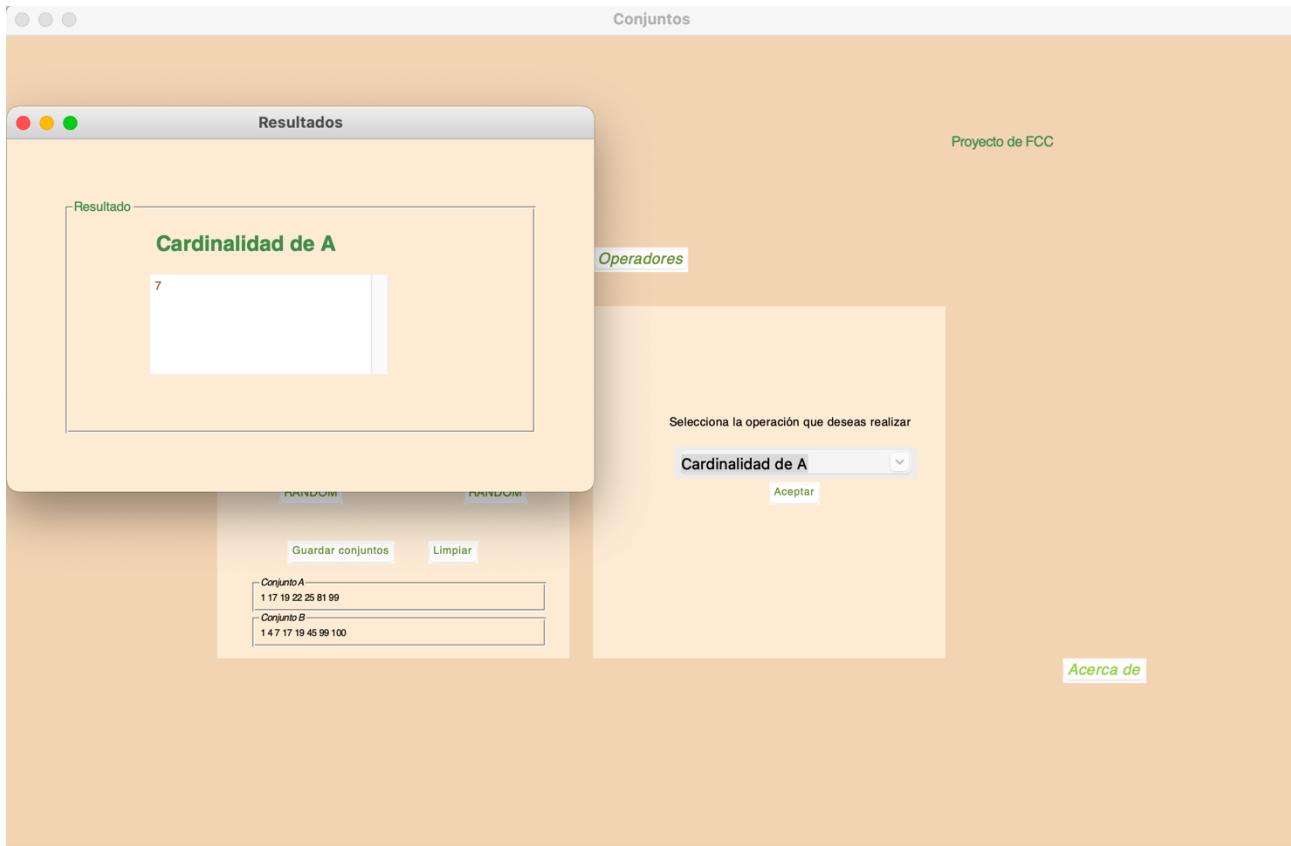
Acerca de

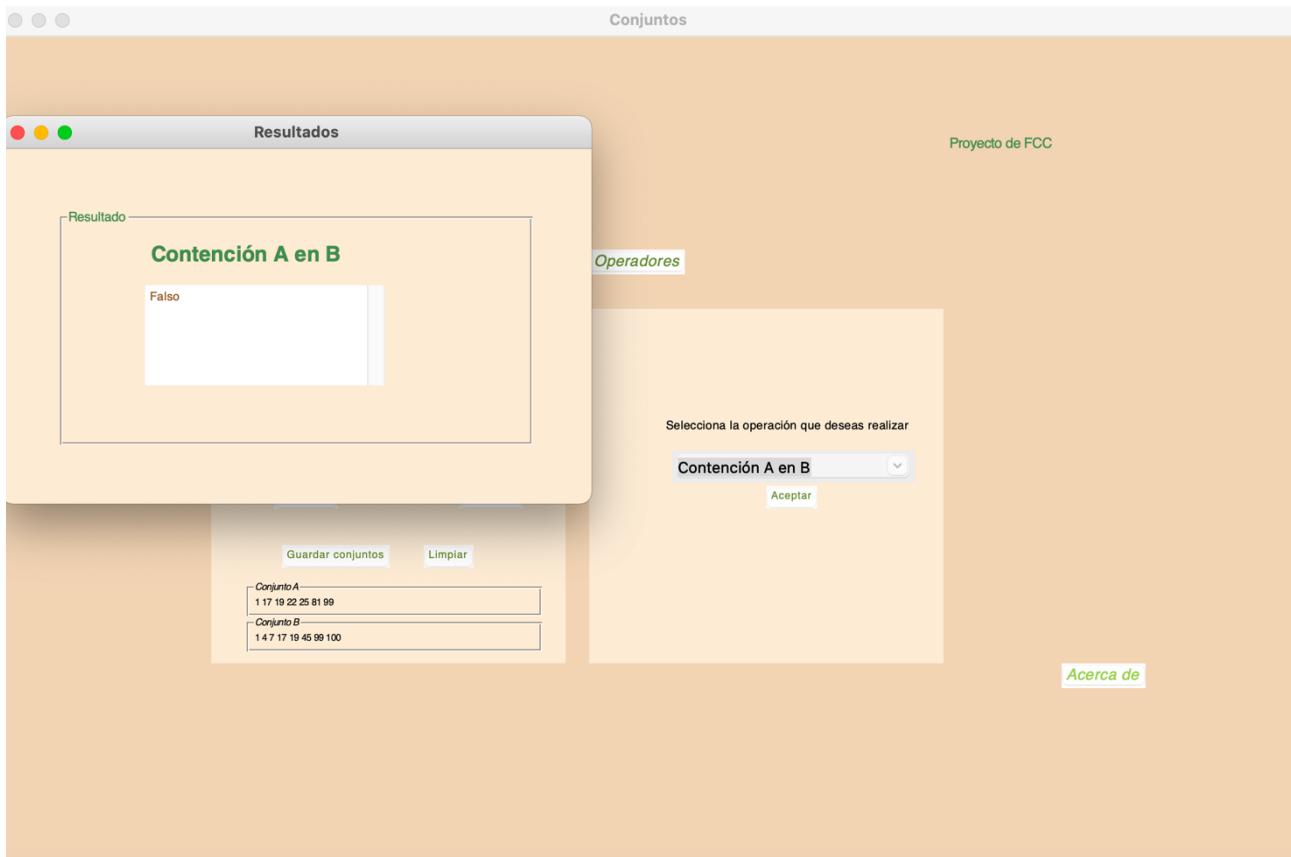


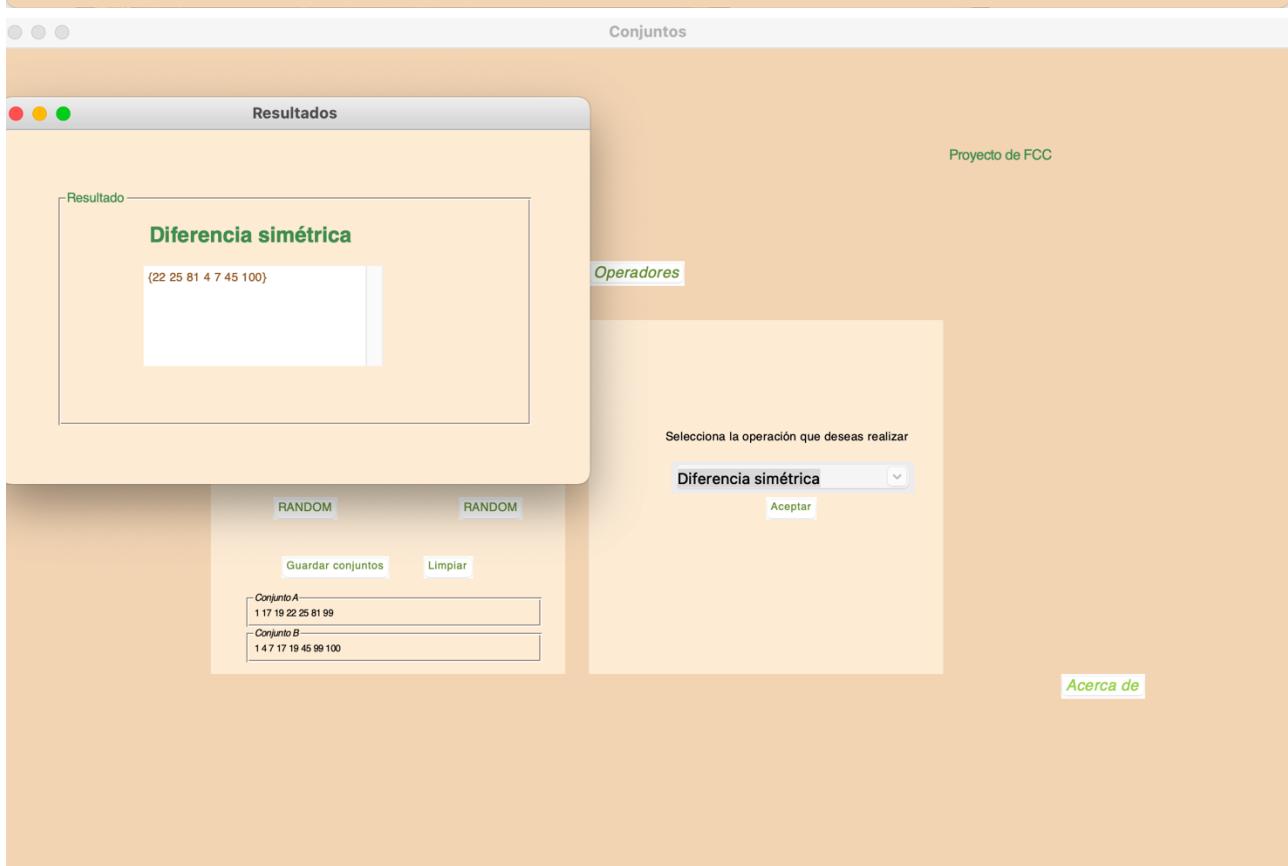
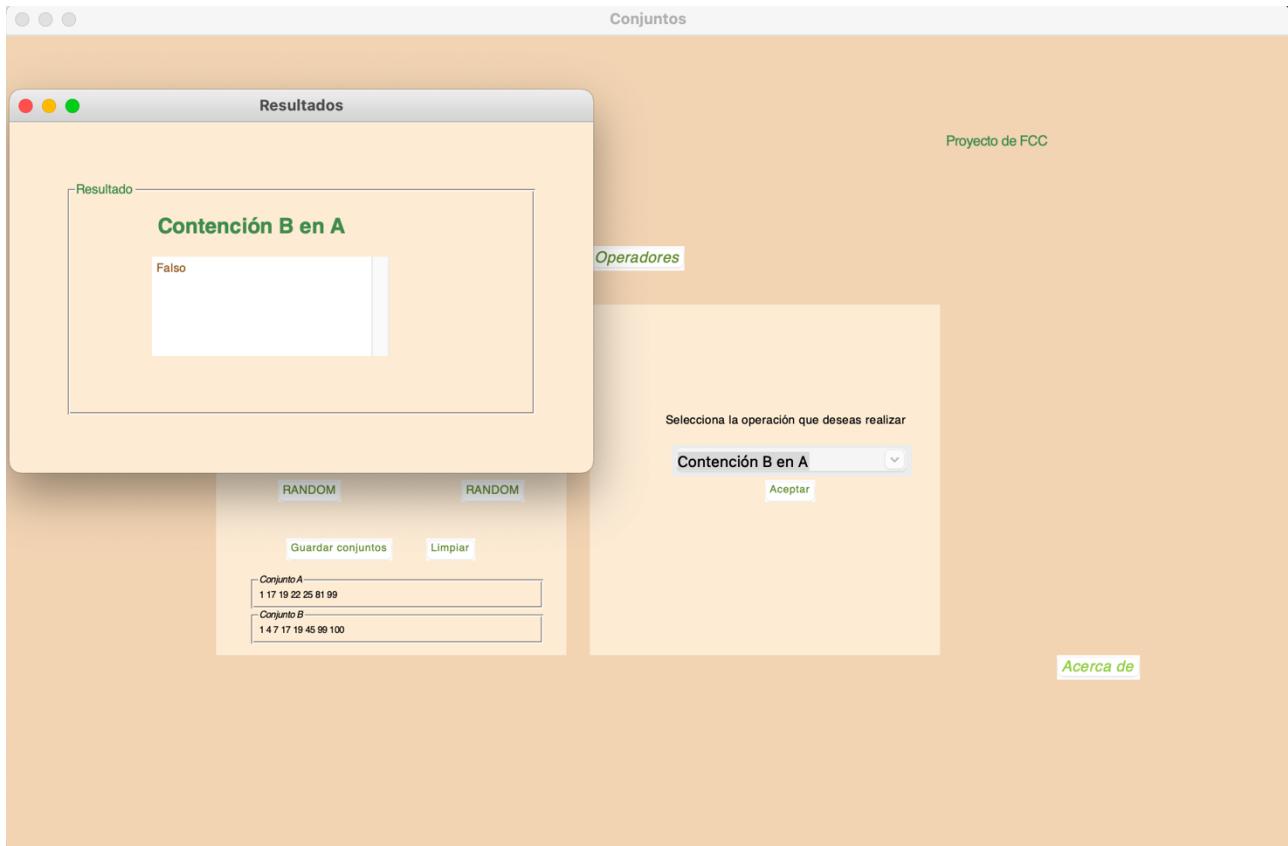












Referencias:

Briega, L. R. E. (2015, 11 octubre). Conjuntos con Python. Matemáticas, Análisis de Datos y Python. Recuperado 25 de abril de 2022, de
<https://relopezbriega.github.io/blog/2015/10/11/conjuntos-con-python/#:%7E:text=El%20conjunto%20potencia%20de%20un,%2C2%2C3%7D%7D>.

Cohaguila, S. (2020, 11 marzo). Operaciones Entre Conjuntos. Ciencias Básicas. Recuperado 25 de mayo de 2022, de
<https://ciencias-basicas.com/matematica/superior/teoria-de-conjuntos/operaciones-entre-conjuntos/>

Oyarzúm, J. (2020). Introducción a la Teoría de Conjuntos. Matemáticas y Ciencias. Recuperado 25 de abril de 2022, de
<http://www.matematicas.ciencias.uchile.cl/juaco/section-2.html>

Tkinter. (2022). tkinter — Interface de Python para Tcl/Tk — documentación de Python - 3.10.4. Python, Tkinter. Recuperado 25 de abril de 2022, de
<https://docs.python.org/es/3/library/tkinter.html>