

**Computational Mathematics**  
Cinvestav-Tamaulipas, Sep – Dec 2022  
Dr. Mario Garza Fabre

**Project 1: Sets**  
**Deadline: 21/10/2022, 12:00pm**

---

## General Specifications

- You will work individually, developing a **Web-based application** that performs set operations.
- You are allowed to use any programming language(s) to develop your application. **However, the logic and code to compute all the set operations must be developed on your own.** That is, you are NOT allowed to use any functionality that your selected programming language may have available for these purposes (yes, in this project you are being asked to reinvent the wheel!).
- Upload all the source codes of your project to your Dropbox folder.
- Also, provide an IP address/URL so that your application can be accessed for evaluation. If required, you will also present your project to the instructor (date/time to be defined).
- Application requirements and the set operations that you will implement are described below. The number of points that the implementation of each operation will contribute to the value of this project is indicated in all the cases for the transparency and objectivity of the evaluation.

## Web Application

- The application will receive two inputs. You could use, for example, the `<textarea>` html tag so that the user can introduce the inputs.
- Tokenisation of the 2 inputs will lead to the construction of 2 sets: A and B. Sets will be formed by the (unique) tokens identified through the analysis of the corresponding inputs (we will refer as “token” to any sequence of characters delimited by spaces, line breaks or punctuation marks).
- Convert inputs to uppercase or lowercase (as you prefer) to deal with case sensitivity.
- As an example, consider the following inputs:

**Input 1:** “Math is fun!”

**Input 2:** “Express the argument in argument form. Determine if the argument is valid.”

Input 1 and Input 2 involve 3 and 12 tokens, respectively. These inputs result in sets A and B with cardinalities 3 and 9, respectively:

**A** = {math, is, fun}

**B** = {express, the, argument, in, form, determine if, is, valid}

- It is possible for one or the two inputs to be empty, this would result in  $A=\{\}$  or  $B=\{\}$ .
- The application must be capable of performing all the (included) operations when the user invokes the start of the computations. **Results must be presented at the same time for all of the operations performed, and the output should be (sufficiently) descriptive in all the cases.**

## Set Operations

The following table lists all the set operations to be implemented and included in the Web application, as well as their corresponding contributions to the evaluation of this project. Set operations are organised in two categories:

- **Unary operations:** Operations on individual sets. Please present results separately for A and B.
- **Binary operations:** Operations involving the two sets A and B.

Unary Operations	Evaluation (points)
Display members of the set	5
Cardinality	5
Power set	15
All possible partitions of the set (enumerate all possible partitions)	Optional (5 extra points in final grade)

Binary Operations	Evaluation (points)
Comparison: $A=B?$	5
Subset: $A\subseteq B?$ $B\subseteq A?$	10
Proper subset: $A\subset B?$ $B\subset A?$	10
Difference: $A-B, B-A$	10
Symmetric diff.: $A\oplus B, B\oplus A$	10
Union: $A\cup B$	10
Intersection: $A\cap B$ Are A and B disjoint?	10
Cartesian product: $A\times B, B\times A$	10

<b>Evaluation (total):</b>	<b>100</b>
----------------------------	------------

**NOTE:** If you prefer, you could develop a **simple program, without the Web interface**. However, only **80%** of the value of this project will be granted (maximum, it depends on the functionalities implemented). The program should be able to be **compiled and run from the command line**.