- Union(S1:Set, S2:Set)→Set: Returns a new set that contains all elements that are in S1, in S2, or in both. Common elements should be included only once.
- Intersection(S1:Set, S2:Set)→Set: Returns a new set that contains only the elements that are in both S1 and S2
- Difference(S1:Set, S2:Set)→Set: Returns a new set that contains the elements in S1 that are not in S2.

Performance Analysis

- There are many criteria upon which we can judge a program, including:
(1) Does the program meet the original specifications of the task?
(2) Does it work correctly? ✓
(3) Does the program contain documentation that shows how to use it and how it works? ✓
(4) Does the program effectively use functions to create logical units?
(5) Is the program's code readable?

More concrete criteria

6) Does the program efficiently use primary and secondary storage?
7) Is the program's running time acceptable for the task?

Defn: The space complexity of a program is the amount of memory that it needs to run to completion. The time complexity of a program is the amount of computer time that it needs to run to completion. □

Space Complexity

- The space needed by a program is the sum of the following components:
(1) **Fixed space requirements:** Do not depend on the number and size of the program's inputs and outputs. Includes the instruction space (space needed to store the code), space for simple variables, fixed-sized structured variables (like structs) and constants.