

Loyola University Chicago
Department of Computer Science
COMP 272: Data Structures II (Fall 2025)
Assignment # 5

This is an individual assignment.

Deadline:

Submission:

1. You need to submit your work electronically by using your assigned assignment located within your Sakai 'Assignment tab'.
2. DO NOT MODIFY THE FILE 'Main.java' on your submitted version
3. Before submission, make sure your **code passes all the supplied tests**. Keep in mind, however, that passing the test cases does not guarantee that your code is correct or efficient. Your assignment will be graded considering test results, correctness, and efficiency.
4. Ensure you include your name and section number as comment at the top the ProblemSolutions.java, BloomFilter.java and CuckooHash.java files.

Assignment:

This assignment contains three short problems which you will develop solutions for. When doing these, utilize the HashMap and PriorityQueue classes within the Java Collection Framework, as appropriate. Two additional Object Classes are provided, BloomFilter and CuckooHash. For these, you will complete the templated methods for these.

Problem Solution - Hash Table (21 points)

Given two arrays A and B, return whether array B is a subset of array A.

Input: [1,50,55,80,90], [55,90]

Output: true

Input: [1,50,55,80,90], [55,90, 99]

Output: false

Problem Solution - Priority Queue 1 (21 points)

Given an array A and integer k, return the kth maximum element in the array.

Input: [1,7,3,10,34,5,8], 4

Output: 7

Problem Solution - Priority Queue 2 (21 points)

Given two arrays A and B with n and m integers respectively, return all the elements in sorted order.

Input: [4,1,5], [3,2]

Output: 1 2 3 4 5

Bloom Filter Hash Map (21 points)

Implement the following methods:

- contains

Complete the contains() method. The method will check the bits in the bloom filter map for each of the 'k' hash codes based on the passed in parameter. It returns false if not in the set, else true if most probably in the set.

Cuckoo Hash Map – (16 points)

Implement the following methods:

- `put`

Complete the `put()` method. This method should add a key-value pair to the table by means of cuckoo hashing.

Note:

- For this **cuckoo hashing problem**, the goal is to intentionally have you study the existing code in order to write the `put()` routine. Do not wait until the last minute to work on this method. Factor in **time management**. This method will likely take additional effort to get it correct and to pass all tests than the previous problems. Even though this may be slightly more difficult than the other problems, the idea is not to penalize too much if your code does not pass the cuckoo hashing tests. That is why this problem is worth the least points.
- Please read the prologue comment for the method ‘`put()`’, including the two HINTs.