Before you start, write your name at the top of each page. This exam was designed to take you 40 minutes or so; you will be given 70. The exam is open-book open-notes. Enough space should be given for each solution, but if not then indicate this and continue on the back.

I suggest that you read the entire exam before you start. If you find a problem with the exam, please note it in your answer and answer as best you can. Please show as much of your work as you reasonably can: I cannot give partial credit for your invisible work.

Note: This is not the real exam. It is just an example to give you an idea of what the real exam might look like.

1. You are given an array \(a\) of \(n\) integers, and another array \(b\) of \(m\) integers. The idea is that \(b\) is an array of “representative” integers for \(a\). That is, each integer in \(a\) is supposedly close to some integer in \(b\) (where by “close” we mean as usual that their absolute difference is small).

   (a) [25 pts] Give pseudocode for a brute-force algorithm to find the largest distance \(d\) from any element in \(a\) to its closest element in \(b\). That is,

   \[
   d = \max_{i \in \{1..n\}} \min_{j \in \{1..m\}}(|a[i] - b[j]|)
   \]

   To find the maximum minimum distance \(d\) from any element in \(a\) to an element in \(b\):

   ```
   d ← 0
   for i in 1..n
       e ← |a[i] - b[0]|
   for j in 1..m
       e ← \min(e, |a[i] - b[j]|)
   d ← \max(d, e)
   return d
   ```

   (b) [25 pts] What is the “big-O” worst-case running time of your algorithm?

   The inner loop executes \(m\) times, and is executed \(n\) times, so the worst-case running time is \(O(nm)\).