linenosize=

Design and Analysis of Computer Algorithms
CS 6363.005: Homework #2

Due on Monday September 26, 2016 at 11:59pm

Professor Benjamin Raichel

Hanlin He (hxh160630)
hanlin.he@utdallas.edu

Contents

Problem 1 Inversions (25 points)

Algorithm 1 Count Inversion of two sorted arrays.

```
1: procedure COUNT_INVERSIONS(A[1...n+m], B[1...n], C[1...m])
2:     InversionsCount = 0
3:     Bindex = 1
4:     Cindex = 1
5:     for Aindex = 1 to n + m do
6:         if Cindex > m then                                  ▷ C is exhausted.
7:             A[Aindex] = B[Bindex]
8:             Bindex++
9:         else if Bindex > n then                             ▷ B is exhausted.
10:            A[Aindex] = C[Cindex]
11:            Cindex++
12:        else if B[index] < C[Cindex] then                   ▷ B is smaller than C.
13:            A[Aindex] = B[Bindex]
14:            Bindex++
15:        else                                    ▷ C is smaller than B, count inversions.
16:            A[Aindex] = C[Cindex]
17:            Cindex++
18:            InversionsCount = InversionsCount + (n − Aindex)
19:        end if
20:    end for
21:    return InversionsCount
22: end procedure
```

Algorithm 2 Sort and Count Inversions

1: **procedure** SORT_AND_COUNT($A[1 \dots n]$)
2: **if** $n > 1$ **then**
3: Sort_and_Count($A[1 \dots \lfloor n/2 \rfloor]$)
4: Sort_and_Count($A[\lfloor n/2 \rfloor + 1 \dots n]$)
5: Count_Inversions($\overline{A}[1 \dots n]$, $A[1 \dots \lfloor n/2 \rfloor$, $A[\lfloor n/2 \rfloor + 1 \dots n]$)
6: