University of Houston

COSC 3320: Algorithms and Data Structures
Spring 2016

Solutions for Homework 1

1. (a) Prove that the sum of the first n even non-negative integers is $n(n-1)$.

   (b) Prove that for every integer $n \geq 0$, $\sum_{i=0}^{n} i^3 = \frac{n^2(n+1)^2}{4}$.

   (c) Prove that for every integer $n \geq 0$, $\sum_{i=0}^{n} i \cdot 2^i = (n-1)2^{n+1} + 2$.

   (d) Prove that for every integer $n \geq 7$, $n^2 \geq 6n + 7$.

   Solution:

   Those claims can be easily proved by induction. Here we shall prove (a); the others can be proved similarly.

   (a) The proof is by induction on n.

   - Base case: $n = 1$. $0 = 1(1-1)$.
   - Inductive step: $n \geq 2$. Suppose, by inductive hypothesis, that the claim is true for all $j < n$. Therefore, for $j = n-1$ we have

   $$\sum_{i=1}^{n-1} 2(i-1) = (n-1)(n-1-1) = (n-1)(n-2).$$

   Hence, we obtain

   $$\begin{aligned}\sum_{i=1}^{n} 2(i-1) &= \sum_{i=1}^{n-1} 2(i-1) + 2(n-1) \\ &= (n-1)(n-2) + 2(n-1) \\ &= (n-1)(n-2+2) \\ &= n(n-1),\end{aligned}$$

   thereby proving the claim.

2. The *element distinctness* problem is the problem of determining whether all the n elements of a list are distinct. Write the pseudocode for the straightforward algorithm that tests each of the n elements for distinctness, and determine its complexity.

   Solution:

   There is an outer loop from $i = 1$ to $n-1$, and then an inner loop from $j = i+1$ to n, inside of which $A[i]$ and $A[j]$ are compared to determine equality: if $A[i] = A[j]$, then return false. Outside the two lops, the instruction will be return true (why? Because if this last instruction will be executed, it means that the instruction "return false" has never been executed, and this means that $A[i] = A[j]$ is never true when i is different than j. Therefore, all the elements are distinct). Below is the pseudocode.

Algorithm 1 Element Distinctness

```
input: a list A = A[1], A[2], ..., A[n]
output: true if all the n elements of A are distinct, false otherwise
for i ← 1 to n − 1 do
    for j ← i + 1 to n do
        if A[i] = A[j] then
            return false
return true
```

Complexity: in the worst case, there are roughly $(n-1)+(n-2)+\cdots+1$ operations, which amounts to $\Theta(n^2)$.

3. (a) Prove that the function $f(n) = 8n + 5$ is $O(n)$.
   (b) Prove that the function $f(n) = 3n^3 + 4n^{5/3} + 2\log n + 8$ is $O(n^3)$.
   (c) Prove that the function $f(n) = 2^{n+2}$ is $\Theta(2^n)$.

   Solution:

   By definition, we just need to exhibit suitable constants c and n_0. The following are possible choices for c and n_0 (but there are infinitely many other correct solutions).

   (a) $c = 9$ and $n_0 = 5$.
   (b) $c = 15$ and $n_0 = 1$.
   (c) $f(n) = O(2^n) : c = 4$ and $n_0 = 0$. $f(n) = \Omega(2^n) : c = 4$ and $n_0 = 0$.