EECS560 **Practice HW#1** **8/31/18**

Instruction: You must show all your work clearly for credit. Partial credit will only be given to meaningful answers.

1. When implementing an ADT for a set of records S, $|S| = 2^6$, it is determined that a find operation, find(x,S), will requires 0.5ms (10^{-3}s) to execute. If the complexity of the find operation is given by the following closed-form expressions T(n), compute the time required to execute this operation when $|S| = 2^{16}$.
   (a) T(n) = 560.
   (b) T(n) = nlgn.
   (c) $T(n) = n^2 \lg n$.
   (d) $T(n) = n^3$.

2. If an algorithm requires 0.5ms to solve a problem with input size of 100, how large a problem it can solve in 1 min if the complexity of the algorithm is given by the following function T(n) in closed-form?
   (a) T(n) = n.
   (b) $T(n) = n^2$.

3. Given the following algorithm for finding the two largest integers in an array A[1..n] of n distinct positive integers. Base on the number of comparisons between elements in A, compute $T_b(n)$ and $T_w(n)$. You must justify your answer and show your work clearly for credit.

```
if A[1] > A[2]                      // Initialization
   then   largest = A[1];
          s_largest = A[2]
   else   largest = A[2];
          s_largest = A[1]
endif;
for i = 3 to n do                   // Checking A[3], ..., A[n]
   if A[i] > s_largest              // A[i] is one of the two largest integers
      then  if A[i] > largest       // A[i] is the current largest integer
                then  s_largest = largest;
                      largest = A[i]
                else  s_largest = A[i]
            endif
   endif
endfor;
```

4. Assuming that all basic operations require the same constant cost C, by concentrating on the dominating step(s), compute the cost of the resource function R(n) for the following program segment in closed-form.

```
x = 2;
y = 10;
for i = 1 to n do
  for  j = i to n do
       y = x * y / 2;
  endfor;
  for k = 1 to n do
       x = x + y − 10;
  endfor;
endfor;
```

5. By concentrating on the dominating step and by assuming that all basic operations require the same constant cost C, compute $T_W(n)$ in closed-form for the following program segment as discussed in class.
   Remark: You must first set up the equation for $T_W(n)$ and then evaluate its sums for credits. Do not simplify the final expression.

```
x = 2;
y = 10;
k = 1;
while k ≤ n do
     x = x + x*y +210;
     y = y – x + 560;
     k = k+1;
endwhile;
for i = 1 to n do
  for  j = i to n do
       y = x * y / 2;
       for k = j to n do
            x = x + y − 10;
       endfor;
  endfor;
endfor;
```

6. Let A_1 and A_2 be two algorithms with closed-form complexity $T_1(n) = 10n^2$ and $T_2(n) = 499n + 50$. Find smallest integer n_0 such that for all $n > n_0$, algorithm A_2 will always be more efficient than algorithm A_1.

7. Use the definition of big-O to prove or disprove that $2^{2n} = O(3^n)$.

8. Prove or disprove that if $T_1(n) = O(f(n))$ and $T_2(n) = O(f(n))$, the $T_1(n) + T_2(n) = O(f(n))$.

9. Prove or disprove that if $T_1(n) = \mathrm{O}(f(n))$ and $T_2(n) = \mathrm{O}(f(n))$, then $\dfrac{T_1(n)}{T_2(n)} = \mathrm{O}(1)$.

10. Using the definition of big-O to prove that $\dfrac{n^4 - n^3 - 2n^2 + 4}{2n^2 - 2n - 27} = \Omega(n^2)$.

11. Use the definition of big-Θ to prove that $\dfrac{2n^4 - n^3 - 5n^2 + 4}{n^2 - 6n + 7} = \Theta(n^2)$.