COMP 4433 Algorithm design and analysis

Midterm Exam 2023W

Open book and notes, no discussion

Q1. (10 points total, 2 points each) Asymptotic function relationships. Verify the following statements by the definitions to see if they are true or false and enter your choices to the D2L:

1. **Q1(1).** If $f(n) = \Theta(n)$, then $nf(n) = \Theta(n^2)$. T
2. **Q1(2).** If $f(n) = O(n^2)$ and $g(n) = \Omega(f(n))$, then $g(n) = O(n^2)$. F
3. **Q1(3).** If $f(n) = O(0.5n^2)$ and $f(n) = \Omega(7n^2 + nlogn)$, then $f(n) = \Theta(n^2)$. T
4. **Q1(4).** If $f(n) = O(n^3)$ then $f(n) = \Omega(2n^3)$. F
5. **Q1(5).** If $f(n) = \Theta(n^3)$ then $f(n) = O(2^n)$. T

Q2 (16 points total: 4 points each, partial points for partially correct) Determine the running time for recursive functions. These are multiple selection questions. Note that $T(1) = \Theta(1)$. Please enter your choices to the D2L:

6. **Q2(1)**. If $T(n) = T(\lceil \frac{n}{2} \rceil) + 1$, then $T(n)$ belongs to a) b) c)
   a) $O(log_2 n)$
   b) $O(n)$
   c) $\Omega(log_2 n)$
   d) $\Omega(n)$
   e) $\Omega(\sqrt{n})$
7. **Q2(2).** If $T(n) = T(n - 1) + n$, then $T(n)$ belongs to c) d) e)
   a) $O(n\, log_2 n)$
   b) $O(n)$
   c) $\Omega(n\, log_2 n)$
   d) $\Omega(n)$
   e) $\Theta(n^2)$

8. **Q2(3)**. If $T(n) = 2T(\lceil \frac{n}{2} \rceil) + \sqrt{n}$, then $T(n)$ belongs to a) b) d)
    a) $O(n\, log_2 n\ )$
    b) $O(n)$
    c) $\Omega(n\, log_2 n\ )$
    d) $\Omega(n)$
    e) $\Theta(n^2)$

9. **Q2(4).** If $T(n) = 16T(\lceil \frac{n}{4} \rceil) + n^2$, then $T(n)$ belongs to c) d)
    a) $O(n\, log_2 n\ )$
    b) $O(n)$
    c) $\Omega(n\, log_2 n\ )$
    d) $\Omega(n)$
    e) $\Theta(n^2)$

Q3. (5 points, see details below and no partial points for each sub-questions) The following algorithm is used to recursively count the number of leaves in a **binary** tree, where T is a binary tree and n is the number of leaves.

```
LeafCounter(T)
        if (T == EmptySet) then
                n = 0
        else
                n = LeafCounter(T.left)+LeafCounter(T.right)
        return n
```

10. **Q3(1).** (2 points) If you think the above algorithm is correct, choose true, otherwise choose false and enter it in the D2L. F

11. **Q3(2).** (3 points) If you think the algorithm is false, please rewrite the pseudocode in D2L.Otherwise, state "N/A" as the answer in the D2L.

```
LeafCounter(T)
        if (T == EmptySet) then
                n = 0
        else if  (T.left == EmptySet and T.right == EmptySet) then
                n=1
        else
                n = LeafCounter(T.left)+LeafCounter(T.right)
        return n
```

Q4. (9 points in total, 3 points each, no partial points) Dynamic Programming questions. The following sub-questions can be multiple selection or single selection questions. Please enter your choices to the D2L:

12. **Q4(1).** The dynamic programming can be used when b) c) d)
    a) it is faster than the Greedy Algorithm.
    b) the solution has an optimal structure.
    c) the solutions to the sub-problems can be combined to give a solution to the original problem.
    d) we want to avoid repeated calculations for sub-problems.

13. **Q4(2).** Given a set of n positive integers, $C = \{c_1, c_2, ..., c_n\}$ and a positive integer K, is there a subset of C whose elements sum to K? A dynamic program for solving this problem uses a 2-dimensional Boolean table T, with n rows and $K + 1$ columns. $T[i, j]$, $1 \le i \le n$, $0 \le j \le K$, is TRUE if and only if there is a subset of $\{c_1, c_2, ..., c_i\}$ whose elements sum to j. Which of the following is valid for $2 \le i \le n$, $c_i \le j \le K$?
    c)

    a) $T[i, j] = (T[i-1, j] \; or \; T[i, j - c_i])$
    b) $T[i, j] = (T[i-1, j] \; and \; T[i, j - c_i])$
    c) $T[i, j] = (T[i-1, j] \; or \; T[i-1, j - c_i])$
    d) $T[i, j] = (T[i-1, j] \; and \; T[i-1, j - c_i])$

14. **Q4(3).** In the above problem (13.Q4(2)) which entry of the table T, if TRUE, implies that there is a subset whose elements sum to K? b)

    a) $T[1, K + 1]$
    b) $T[n, K]$
    c) $T[n, 0]$
    d) $T[n, K + 1]$