COM S 311 SPRING 2022
HOMEWORK 1

Due: February 10, 11:59 p.m.

Late Submission Due: February 11, 11:59 A.M. (25% penalty)

(1) In each of the following situations, indicate whether $f(n) = O(g(n))$, or $f(n) = \Omega(g(n))$, or both (in which case $f(n) = \Theta(g(n))$). No proof is required.

	$f(n)$	$g(n)$
(a)	$2n + 300$	$100n + 2$
(b)	$n^{3/4}$	$n^{2/3}$
(c)	$10n^2 + (\log n)^3$	$n + \log n$
(d)	$37n \log(10^6 n)$	$n \log n$
(e)	$n^2 \log n$	$n^{2.0001}(\log n)^2$
(f)	$(\log n)^2$	$n/\log n$
(g)	$n/\log n$	$(\log n)^{\log n}$
(h)	$(\log n)^3$	$\sqrt[3]{n}$
(i)	$3^{\log_2 n}$	$n^{\log_2 3}$
(j)	3^n	$n^4 2^n$
(k)	$n!$	2^n
(l)	$2^{(\log n)^2}$	$(\log n)^{\log n}$

(2) Solve the following problems using the formal definitions of O, Ω, and Θ.[1]
  (a) Show that $\sum_{i=1}^{n} i^2$ is $O(n^3)$.
  (b) Prove or disprove: $\sum_{i=1}^{n} i^2$ is $\Omega(n^3)$.
  (c) Show that for any real constants, where $b > 0$, $(n+a)^b = \Theta(n^b)$. Prove this by showing that
    (i) $(n + a)^b = O(n^b)$ and
    (ii) $(n + a)^b = \Omega(n^b)$.

(3) Solve the following problems using the formal definitions of O.
  (a) Show that $5n^2 \log n + 3n^2 + 7n \log n + 9n + 2$ is in $O(n^2 \log n)$.
  (b) Show that $3n^2(\sqrt{n}) + 5n \log n + 7$ is not in $O(n^2)$.

(4) Consider the following algorithm.

[1] By "using the formal definitions", we mean, for example, that to show that $f(n)$ is $O(g(n))$ you must show that there exist positive constants c and n_0 such that $f(n) \leq c \cdot g(n)$, for all $n \geq n_0$. Similarly for Ω and Θ.

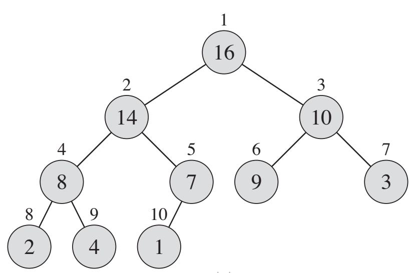
```
SubTotals(A)
    Input: An array A of n integers.
    Output: An n × n array B, where B[i, j] = 0 if i > j, and B[i, j] = Σ_{k=i}^{j} A[k] otherwise.
    for i = 1 to n do
        for j = 1 to n do
            B[i, j] = 0
    for i = 1 to n do
        for j = i to n do
            sum = 0
            for k = i to j do
                sum = sum + A[k]
            B[i, j] = sum
    return B
```

(a) What is the running time of SubTotals as a function of n? Justify your answer.

(b) Is your bound for part (a) tight? (That is, is your bound a Θ-bound, not just a O-bound?) Justify your answer.

(c) Design an algorithm that produces the same output as SubTotals, but is asymptotically faster. That is, if $T_{\texttt{SubTotals}}(n)$ is the running time of SubTotals (from part (a)) and $T_{\text{new}}(n)$ is the running time of your new algorithm, then $\lim_{n\to\infty} T_{\text{new}}(n)/T_{\texttt{SubTotals}}(n) = 0$. Justify your answer.

(5) Design an algorithm that takes as input an n-element heap H and a key k and returns all the entries in H having a key greater than or equal to k. For example, given the heap below and query $k = 8$, the algorithm should report the entries with keys $16, 14, 10, 8,$ and 9 (but not necessarily in this order). Your algorithm should run in time proportional to the number r of entries returned, and should not modify the heap. You may assume that at least one entry will be returned. Note that r can be much smaller than n.



Source: CLRS

For full credit,

- describe your algorithm using pseudocode,
- argue the correctness of your algorithm, and
- analyze the running time of your algorithm (that is, explain why it runs in $O(r)$ time).

Guidelines

- It is important to know whether you really know. For each problem, if you write the statement "I do not know how to solve this problem" (and nothing else), you will receive 20% credit for that problem. If you do write a solution, then your grade could be anywhere between 0% to 100%. To receive this 20% credit, you must explicitly state that you do not know how to solve the problem.

- You are allowed to discuss homework with classmates, but you must write the final solutions alone, without consulting anyone. Your writing should demonstrate that you completely understand the solution you present.
- When proofs are required, you should make them both clear and rigorous. Do not hand-waive.
- Please submit your assignment via Canvas.
  - If you type your solutions, then please submit a PDF version.
  - If you hand-write your solutions, then please scan or take a picture of your solutions and submit a PDF version. Please make sure that the quality of the scan/picture is good, and that your handwriting is legible.
  - Please make sure that the file you submit is not corrupted and that its size is reasonable (e.g., roughly at most 10-11 MB).

    If we cannot open your file, your homework will not be graded.
- Any concerns about grading should be expressed within one week of returning the homework.

Note: We reserve the right to grade only a subset of the problems assigned. Which problems will be graded will be decided after the submission deadline.