CMPSC 465 Data Structures & Algorithms
Spring 2021 Paul Medvedev and Chunhao Wang

HW 2

Due February 4, 10:00 pm

Instructions: You may work in groups of up to three people to solve the homework. You must write your own solutions and explicitly acknowledge up everyone whom you have worked with or who has given you any significant ideas about the HW solutions. You may also use books or online resources to help solve homework problems. All consulted references must be acknowledged.

You are encouraged to solve the problem sets on your own using only the textbook and lecture notes as a reference. This will give you the best chance of doing well on the exams. Relying too much on the help of group members or on online resources will hinder your performance on the exams.

Late HWs will be accepted until 11:59pm with a 20% penalty. HWs not submitted by 11:59pm will receive 0. There will be no exceptions to this policy, as we post the solutions soon after the deadline. However, you will be able to drop the three lowest HW grades.

For the full policy on HW assignments, please consult the syllabus.

1. (0 pts.) Acknowledgements. The assignment will receive a 0 if this question is not answered.

(a) If you worked in a group, list the members of the group. Otherwise, write "I did not work in a group."

(b) If you received significant ideas about the HW solutions from anyone not in your group, list their names here. Otherwise, write "I did not consult without anyone my group members".

(c) List any resources besides the course material that you consulted in order to solve the material. If you did not consult anything, write "I did not consult any non-class materials."

2. (30 pts.) Compare Growth Rates. In each of the following situations, indicate whether $f = O(g)$, or $f = \Omega(g)$, or both (in which case $f = \Theta(g)$). Give a one sentence justification for each of your answers.

	$f(n)$	$g(n)$
(a)	$n^{1.5}$	$n^{1.3}$
(b)	2^{n-1}	2^n
(c)	$n^{1.3\log n}$	$n^{1.5}$
(d)	3^n	$n2^n$
(e)	$(\log n)^{100}$	$n^{0.1}$
(f)	n	$(\log n)^{\log\log n}$
(g)	2^n	$n!$
(h)	$\log(e^n)$	$n\log n$
(i)	$n+\log n$	$n+(\log n)^2$
(j)	$5n+\sqrt{n}$	$\log n+n$

3. (20 pts.) Polynomials and Horner's rule (HW) Suppose we want to evaluate the polynomial $p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n$ at some point x_0.

(a) Consider the brute force algorithm that computes first a_1x_0, $a_2x_0^2$, $a_3x_0^3, \ldots, a_nx_0^n$ (independently) and then adds all of these numbers to a_0 to obtain $p(x_0)$. How many sums and how many multiplications are involved? (You can give your answer in O-notation.)

(b) We show next how to do better using *Horner's rule*. Prove that the following algorithm outputs $p(x_0)$. Use the loop invariant proof structure we saw in class.

```
z = a_n;
for i = n − 1 to 0 do
|   z = z · x_0 + a_i;
end
return z;
```

(c) How many additions and multiplications does this algorithm use as a function of n? (You can give your answer in O-notation.)

4. (30pt pts.) Solving recurrences Solve the following recurrence relations and give a Θ bound for each of them. Do not use the Master Theorem. You must use the recursion tree method. For each recurrence, make sure you state the branching factor, the height of the tree, the size of the subproblems at depth k, and the number of subproblems at depth k. It is not necessary to draw the tree, though you may do so anyway to help yourself visualize the problem.

(a) $T(n) = 2T(n/2) + \sqrt{n}$

(b) $T(n) = 2T(n/3) + 1$

(c) $T(n) = 5T(n/4) + n$

(d) $T(n) = 7T(n/7) + n$

(e) $T(n) = 9T(n/3) + n^2$