

# Foundations of Computer Science HW 5

Greg Stewart

March 6, 2018

## Q1) 8.7(d)-(f)

*Problem.* The set  $P$  of parenthesis strings has a recursive definition as shown:

$$\begin{aligned}\epsilon &\in P \\ x \in P &\implies [x] \in P \\ x, y \in P &\implies xy \in P\end{aligned}$$

- (e) Prove by structural induction that every string in  $P$  is balanced.

Consider the base case, the empty string. There are 0 [ and 0 ], so the string is balanced. Now for the first constructor, let the string  $x$  be balanced. Then the constructor constructs a string with 1 more [ and 1 more ], which obviously maintains the balance of the string. For the second constructor, let  $x$  and  $y$  both be balanced strings. By counting the brackets as in part (f), this means  $x$  and  $y$  both have balance 0, so when they are concatenated, they have a balance of  $0+0=0$ . Thus the second constructor also produces balanced strings, and every string in  $P$  is balanced.

- (f) For a string  $x \in P$ , define the imbalance as follows: start on the left of  $x$  and move right; add +1 for every [ you encounter, and add -1 for every ] you encounter.

- (i) After you traverse  $x$ , what is the imbalance?

The imbalance is 0 after traversing  $x$ .

- (ii) Give an upper bound on the imbalance at any point in  $x$ .

At most, the imbalance can be  $n$ , where  $n$  is the number of total recursions. This means the imbalance can also be defined as at most half the length of  $x$ :  $\frac{1}{2} \cdot \text{length}(x)$ .

- (iii) Prove by structural induction that at any point in  $x$ , imbalance  $\geq 0$ .

Consider a string  $x$  in  $P$ . By the first constructor, any ] must be preceded by a complementary [. For the empty string, there is only one point in the traversal, and the imbalance is of course 0. Using the first constructor, if we assume that at any point in a traversal of  $x$ , imbalance  $\geq 0$ , then when a new string is constructed,  $[x]$ , we have the same thing. This is because 1 is added to the imbalance at the beginning, and not subtracted until the very end of the traversal, so during the entire traversal, imbalance  $\geq 0$  must be true. Now assume that imbalance  $\geq 0$  for both strings  $x$  and  $y$  in  $P$ . Then when traversing the concatenation  $xy$ , we can think of the traversal in two parts: first over the  $x$  portion, then over the  $y$  portion. As the imbalance is greater than or equal to 0 for each  $x$  and  $y$  separately, then at worst the imbalance can be  $0+0=0$  at some point in the traversal; otherwise it must be positive, proving that imbalance  $\geq 0$  for any point in string  $x \in P$ .

- (g) In the text, we defined the  $M$  of balances and matched parentheses. Prove that  $P = M$ :

$$\begin{aligned}\epsilon &\in M \\ x, y \in M &\implies [x] \bullet y \in M\end{aligned}$$

- (i) *Prove by structural induction that every  $x \in P$  has a derivation using the rules for  $M$ .*

The base case for  $P$  is the same as that of  $M$ , so we have that. Now consider the first constructor of  $P$ . For every  $x$  in  $P$ , we can make a new string  $[x]$ . Assume that the same  $x$  exists in  $M$ , and let  $y = \epsilon$ . Then  $[x] \in P$  can be constructed using the constructor for  $M$  since  $[x] \bullet \epsilon = [x]$ . Now take the second constructor that produces strings in  $P$ : it concatenates any strings  $x$  and  $y$  in  $P$ , giving  $xy$ . To produce this string using the constructor of  $M$ , let  $y \in M$  be the same string as  $y \in P$ , and let there be a string  $z \in M$  such that  $[z] = x \in P$ , which of course can be constructed from the first constructor of  $P$ . Then from  $M$ 's constructor, we have  $[z] \bullet y = xy \in M$ . Thus every string  $x \in P$  has a derivation using the rules of  $M$ .

- (ii) *Prove by structural induction that every  $x \in M$  has a derivation using the rules for  $P$ .*

As before, the base cases are the same for  $M$  and  $P$ , so we have that. Now consider the one constructor that makes  $M$ . Take some  $x \in M$  and let  $y = \epsilon$ . Then the constructor constructs  $[x] \bullet \epsilon = [x] \in M$ ; letting  $x \in P$  be the same string, we construct from the first constructor of  $P$  the string  $[x] \in P$ , so  $[x] \in (P \wedge M)$ , and this set of strings can be derived from the rules of  $P$ . Now assume that  $x$  and  $y$  are both nonempty strings in  $M$ . Then we construct from the rule of  $M$  the string  $[x]y \in M$ . Assume the same strings  $x$  and  $y$  are in  $P$ . Then from the first constructor we can derive the string  $[x]$ , and from the second we know  $[x], y \implies [x]y \in P$ , so we have produced the same string,  $[x]y$ , using the rules of  $P$ . Thus, every string in  $M$  can be derived using the rules of  $P$ .

As both these cases can be proved, it follows that  $P = M$ .

## Q2) 8.20(a)-(b)

*Problem.* Recursively define the binary strings that contain more 0's than 1's.

Base case.  $0 \in P$

$$\begin{aligned} x \in P &\implies 0x \in P \\ x \in P &\implies x0 \in P \\ x \in P &\implies 0x1 \in P \\ x \in P &\implies 1x0 \in P \\ x \in P &\implies x01 \in P \\ x \in P &\implies 10x \in P \\ x \in P &\implies 01x \in P \\ x \in P &\implies x10 \in P \end{aligned}$$

- (a) *Prove that every string in your set has more 0's than 1's.*

This is pretty simple by induction. We have the base case, 0, which obviously has more 0's than 1's, since 1 is greater than 0. Consider each of the directions the string can go in. Say we have a string in  $P$  with  $g$  0's and  $r$  1's such that  $g > r$ . Then for the next step, either one 0 and one 1 is added, or only one 0 is added. In the former,  $g+1 > r+1$  so the string still has more 0's than 1's. In the latter,  $g+1 > r$ , and the rule is still maintained. So every string has more 0's than 1's.

- (b) *Prove that every string which has more 0's than 1's is in your set.*

Say that there is some binary string  $s$  which is the smallest string with more 0's than 1's that is not in  $P$ . If  $s$  starts with 1, then either  $s = 1 \bullet x \bullet 0$  or  $s = 1 \bullet 0 \bullet x$ . This means  $x$  must have more 0's than 1's for  $s$  to have more 0's than 1's. Since  $s$  is the smallest string of this kind not in  $P$ , we must have  $x \in P$ , but this means that by those constructor rules,  $s \in P$ , so we have a contradiction.

Likewise, if  $s$  starts with 0, we could have  $s = 0 \bullet x$ ,  $s = 0 \bullet x \bullet 1$ , or  $s = 0 \bullet 1 \bullet x$ . In any case, we see that  $x$  must have more 0's than 1's, so  $x \in P$ . Then by the constructor rules,  $s \in P$ . Another contradiction.

This is again true for  $s$  starting and ending with 0, starting and ending with 1, starting with 1 and ending with 0, and starting with 0 and ending with 1. In all cases, we get the same contradiction. Therefore there is no shortest binary string with more 0's than 1's not in  $P$ , and  $P$  contains all such strings.

### Q3) 9.2(g)-(h)

*Problem.* Tinker and then compute formulas that do not contain a sum for the following:

(g)  $\sum_{i=1}^n ij = j \sum_{i=1}^n i = j \cdot \frac{1}{2}n(n+1)$

(h)  $\sum_{i=0}^n (i+j)^2$

$$\begin{aligned}
\sum_{i=0}^n (i+j)^2 &= \sum_{i=0}^n (i^2 + 2ij + j^2) \\
&= \sum_{i=0}^n i^2 + \sum_{i=0}^n 2ij + \sum_{i=0}^n j^2 \\
&= \left[ \frac{1}{6}n(n+1)(2n+1) + 0 \right] + 2j \sum_{i=0}^n i + (n+1)j^2 \\
&= \left[ \frac{1}{6}n(n+1)(2n+1) + 0 \right] + 2j \frac{1}{2}n(n+1) + (n+1)j^2 \\
&= \frac{1}{6}(n^2+n)(2n+1) + (n^2+n)j + (n+1)j^2
\end{aligned}$$

### Q4) 9.3(f)-(h)

*Problem.* Compute formulae that do not contain a sum for the following:

(f)  $\sum_{i=0}^n \sum_{j=i}^n (1+j)$  or  $\sum_{i=0}^n \sum_{j=i}^n (i+j)$ ; I did both because I wasn't sure.

$$\begin{aligned}
\sum_{i=0}^n \sum_{j=i}^n (1+j) &= \sum_{i=0}^n \sum_{j=i}^n 1 + \sum_{i=0}^n \sum_{j=i}^n j \\
&= \sum_{i=0}^n (n+1-i) + \sum_{i=0}^n \left( \sum_{j=0}^n j + \sum_{j=0}^{i-1} j \right) \\
&= \sum_{i=0}^n \left( n+1 - i \frac{n^2}{2} + \frac{n}{2} - \frac{i^2}{2} - \frac{i}{2} \right) \\
&= \sum_{i=0}^n \left( \frac{n^2}{2} + \frac{3n}{2} - \frac{i^2}{2} - \frac{i}{2} + 1 \right) \\
&= (n+1) \frac{n^2}{2} + (n+1) \frac{3n}{2} + (n+1) - \frac{1}{2} \frac{1}{6} n(n+1)(2n+1) - \frac{1}{2} \frac{1}{2} n(n+1) \\
&= \frac{n^3}{2} + \frac{n^2}{2} + \frac{3n^2}{2} + \frac{3n}{2} + n + 1 - \frac{1}{12} (2n^3 + n^2 + 2n^2 + n) - \frac{1}{4} (n^2 + n) \\
&= \frac{n^3}{3} + \frac{3n^2}{2} + \frac{13n}{6} + 1
\end{aligned}$$

(g)  $\sum_{i=0}^n \sum_{j=0}^i 2^i$

$$\begin{aligned}
\sum_{i=0}^n \sum_{j=0}^i 2^i &= \sum_{i=0}^n 2^i \sum_{j=0}^i 1 \\
&= \sum_{i=0}^n 2^i \cdot (i+1) = \sum_{i=0}^n i 2^i + \sum_{i=0}^n 2^i \\
&= (0 + 1 \cdot 2 + 2 \cdot 2^2 + \cdots + (n-1) \cdot 2^{n-1} + n \cdot 2^n) + 2^{n+1} - 1 \\
&= n \cdot 2^{n+1} - 2^{n+1} + 2 + 2^{n+1} - 1 \\
&= n 2^{n+1} + 1
\end{aligned}$$

(h)  $\sum_{i=0}^n \sum_{j=0}^i j \cdot 2^i$

$$\begin{aligned}
\sum_{i=0}^n \sum_{j=0}^i j \cdot 2^i &= \sum_{i=0}^n 2^i \sum_{j=0}^i j \\
&= \sum_{i=0}^n 2^i \cdot \frac{1}{2} i(i+1) \\
&= \sum_{i=0}^n \left( \frac{i^2 2^i}{2} + \frac{i \cdot 2^i}{2} \right) \\
&= \frac{1}{2} \sum_{i=0}^n i^2 2^i + \frac{1}{2} \sum_{i=0}^n i 2^i \\
&= \frac{1}{2} (2^{n+1} n^2 - 2^{n+2} n + 3 \cdot 2^{n+1} - 6 + 2^{n+1} n - 2^{n+1} + 2) \\
&= 2^n n^2 - 2^{n+1} n + 3 \cdot 2^n - 3 + 2^n n - 2^n + 1 \\
&= 2^n n^2 - 2^{n+1} n + 2^n n + 2^{n+1} - 2 \\
&= 2^n (n^2 + n) + 2^{n+1} (1 - n) - 2
\end{aligned}$$

## Q5) 9.15(a)-(e)

*Problem.* Say whether the following are equal to  $\Theta(n)$ ,  $\Theta(n^2)$  or neither.

- (a)  $10 \neq \Theta(n)$  or  $\Theta(n^2)$
- (b)  $3n + 9 = \Theta(n)$  but not  $\Theta(n^2)$
- (c)  $\lfloor n \rfloor = \Theta(n)$  but not  $\Theta(n^2)$
- (d)  $\lceil n/2 \rceil = \Theta(n)$  but not  $\Theta(n^2)$
- (e)  $n^2 + n + 1 = \Theta(n^2)$  but not  $\Theta(n)$

## Q6) 9.23(b)

*Problem.* Prove by contradiction:  $2^n \notin \Theta(3^n)$

Say that  $2^n = \Theta(3^n)$ . Then when we take the limit of  $\frac{3^n}{2^n}$ , we should obtain a nonzero constant. Let's try this:

$$\lim_{n \rightarrow \infty} \frac{3^n}{2^n} = \lim_{n \rightarrow \infty} \left( \frac{3}{2} \right)^n$$

which we can write as

$$\left(\frac{3}{2}\right)^{\lim_{n \rightarrow \infty} n} = \left(\frac{3}{2}\right)^{\infty} = \infty$$

and since this limit is  $\infty$ , we have that  $2^n = O(3^n)$  and  $3^n \neq O(2^n)$ , which means we have a contradiction with our initial assumption.

Then we should have  $2^n \leq 3^n$  and  $3^n \leq 2^n$ . For the  $n = 0$  case, this is true for both of these. Assume it's true for some  $n$ . Since both of these inequalities must be satisfied, we can require

$$2^n = 3^n.$$

Consider the  $n + 1$  case: we have

$$\begin{aligned} 2^{n+1} &= 3^{n+1} \\ 2 \cdot 2^n &= 3 \cdot 3^n \\ 2^n &= \frac{3}{2} \cdot 3^n \end{aligned}$$

But we should have  $2^n = 3^n$ , not  $\frac{3}{2} \cdot 3^n$ , so this contradicts our assumption! Thus we cannot have that  $2^n \in O(3^n)$  and  $3^n \in O(2^n)$ , and therefore

$$2^n \notin \Theta(3^n).$$

## Q7) 9.48(a)-(b)

*Problem.* Refer to the following algorithm:

input: Array [a<sub>1</sub>, ..., a<sub>k</sub>]

```

1  MaxSum := 0
2  for i = 1..n
3      for j = i..n
4          CurSum := 0
5          for k = i..j
6              Cursum := CurSum + a_k
7          MaxSum := max(CurSum, MaxSum)
8  return MaxSum

```

(a) Assume every operation takes 1 unit of time. Show the running time  $T(n)$  is given by

$$T(n) = 2 + \sum_{i=1}^n \left[ 2 + \sum_{j=i}^n \left( 5 + \sum_{k=i}^j 4 \right) \right]$$

On line 1, the assignment takes 1 operation, and the final return operation also takes 1 unit, so we have a constant 2 out front. The rest of the algorithm, from lines 2 through 7, occurs over an interval, iterating from 1 to  $n$ . For each iteration, the variable  $i$  is assigned, then compared to  $n$  to see if it's time to break out of the loop. So far we have:

$$T(n) = 2 + \sum_{i=1}^n \left[ 2 + \text{something} \right]$$

Next, there is a nested for loop iterating  $j$  from  $i$  to  $n$ , which involves five operations, plus another nested loop: assignment of  $j$ , comparing  $j$  to  $n$ , assigning  $CurSum$ , comparing  $CurSum$  to  $MaxSum$ , and

assigning *MaxSum*, plus the nested loop. The nested loop has four operations: assigning  $k$ , comparing  $k$  to  $j$ , adding *CurSum* with  $a_k$ , and assigning *CurSum*. So finally, we have the running time

$$T(n) = 2 + \sum_{i=1}^n \left[ 2 + \sum_{j=i}^n \left( 5 + \sum_{k=i}^j 4 \right) \right]$$

(b) Evaluate the triple nested sum to show that  $T(n) = 2 + \frac{35}{6}n + \frac{9}{2}n^2 + \frac{2}{3}n^3$ .

$$\begin{aligned} T(n) &= 2 + \sum_{i=1}^n \left[ 2 + \sum_{j=i}^n \left( 5 + \sum_{k=i}^j 4 \right) \right] \\ &= 2 + \sum_{i=1}^n \left[ 2 + \sum_{j=i}^n \left( 5 + 4 \sum_{k=i}^j 1 \right) \right] \\ &= 2 + \sum_{i=1}^n \left[ 2 + \sum_{j=i}^n (5 + 4j + 4 - 4i) \right] \\ &= 2 + \sum_{i=1}^n \left[ 2 + (9 - 4i) \sum_{j=i}^n 1 + 4 \sum_{j=i}^n j \right] \\ &= 2 + \sum_{i=1}^n \left[ 2 + (9 - 4i)(n + 1 - i) + 4 \left( \sum_{j=0}^n j - \sum_{j=0}^{i-1} j \right) \right] \\ &= 2 + \sum_{i=1}^n \left[ 2 + (9 - 4i)(n + 1 - i) + 4 \left( \frac{n(n+1)}{2} - \frac{i(i-1)}{2} \right) \right] \\ &= 2 + \sum_{i=1}^n \left[ 2 + 9(n+1) - 9i - 4i(n+1) + 4i^2 + 2n^2 + 2n - 2i^2 + 2i \right] \\ &= 2 + (2 + 9(n+1) + 2n^2 + 2n) \sum_{i=1}^n 1 + (-9 - 4(n+1) + 2) \sum_{i=1}^n i + 2 \sum_{i=1}^n i^2 \\ &= 2 + (11 + 11n + 2n^2)n + (-11 - 4n) \frac{n(n+1)}{2} + \frac{1}{3}n(n+1)(2n+1) \\ &= 2 + 11n + 11n^2 + 2n^3 - \frac{4n^3 + 15n^2 + 11n}{2} + \frac{1}{3}(2n^3 + 3n^2 + n) \\ &= 2 + \frac{35}{6}n + \frac{9}{2}n^2 + \frac{2}{3}n^3 \end{aligned}$$

## Q8) 10.4(a)

*Problem.* Use Euclid's algorithm and the remainders generated to solve the problem: Compute  $\gcd(2250, 1200)$  and find  $x, y \in \mathbb{Z}$  for which  $\gcd(2250, 1200) = 2250x + 1200y$ .

$a$	$b$	$r = a \bmod b$	combination
2250	1200	1050	2250 - 1200
1200	1050	150	1200 - 1050
1050	150	0	1050 - 7(150)

So  $\gcd(2250, 1200) = 150$ .

We can work backwards to get the linear combination of the numbers which produces the gcd:

$$\begin{aligned}
150 &= 1200 - 1050 \\
&= 1200 - (2250 - 1200) \\
&= -2250 + 2(1200)
\end{aligned}$$

So  $x = -1$  and  $y = 2$ .

### Q9) 10.8(a)

*Problem.* Prove that consecutive integers  $n$  and  $n + 1$  are relatively prime.

Consider the greatest common divisor of  $n$  and  $n + 1$ . If we find this in the standard, Euly way, then we get it almost immediately:

$a$	$b$	$r = a \bmod b$	combination
$n + 1$	$n$	1	$n + 1 - n$
$n$	1	0	$n - n \cdot 1$

So  $\gcd(n+1, n) = 1$ . Since the greatest common divisor of these adjacent numbers is 1, they are relatively prime.

### Q10) 10.27(b)

*Problem.* Use modular arithmetic. What is the last digit of  $3^{2016} + 4^{2016} + 7^{2016}$ ?

This is as simple as finding the remainder when divided by 10, i.e. the sum mod 10.

$$\begin{aligned}
3^{2016} + 4^{2016} + 7^{2016} &\equiv (3^4)^{504} + (4^4)^{504} + (7^4)^{504} \pmod{10} \\
&\equiv 81 + 256 + 2401 \equiv 2738 \pmod{10} \\
&\equiv 8 \pmod{10}
\end{aligned}$$

So the last digit of this number is 8.

### Q11) 11.3

*Problem.* A graph is regular if every vertex has the same degree. Say whether each of these is regular:

- $K_6$  is regular.
- $K_{4,5}$  is not regular.
- $K_{5,5}$  is regular.
- $L_6$  is not regular.
- $S_6$  is not regular.
- $W_4$  is regular.
- $W_5$  is not regular.

### Q12) 11.8(a)-(c)

*Problem.* Compute the number of edges in the following graphs:

(a)  $K_n$

Every node is connected to every other node, so each node has  $n - 1$  edges coming out of it. There are  $n$  nodes, so we get  $n(n - 1)$  edges.

(b)  $K_{n,l}$

This bipartite graph has  $n$  nodes each connected to  $l$  nodes, so the total number of edges is  $nl$ .

(c)  $W_n$

A cycle graph has  $n$  nodes and as each node has 2 connections, there are  $n$  edges. When a center vertex is added to make this a wheel, we get another  $n$  edges. So in total there are  $n + n = 2n$  edges.