

TCSS 343 - Week 0 - Thursday

Jake McKenzie

December 19, 2018

Recursion and some Mathematical Review

“So plant your own gardens and decorate your own soul, instead of waiting
for someone to bring you flowers.”

...

Jorge Luis Borges

“It’s known in theory that $\log \log n$ approaches infinity,
but no one has ever observed it in practice.”

...

Grant Sanderson (3blue1brown on youtube)

Listen to the **MUSTN'TS**, child,
Listen to the **DON'TS**
Listen to the **SHOULDN'TS**
The **IMPOSSIBLES**, the **WONT'S**
Listen to the **NEVER HAVES**
Then listen close to me-
Anything can happen, child,
ANYTHING can be

...

Shel Silverstein

0. For the following problem I want you to trace the code. What is the final value that is printed? To accomplish this, *correctly*, you will need to keep track of each the stack trace for each recursive call. That means keeping track of the values of x, y, u, v for 6 recursive calls.

```
1  def debug_week0_thursday(x,y,u,v):
2      if x > y:
3          x = x - y;
4          u = u + v;
5          debug_week0_thursday(x,y,u,v);
6      elif y > x:
7          y = y - x;
8          v = v + u;
9          debug_week0_thursday(x,y,u,v);
10     else:
11         print((x + y) / 2 * (u + v) / 2);
12     debug_week0_thursday(36,10,36,10)
```

1. For this problem we will prove the proposition $P(n) : \mathcal{F}(n) < 2^n$ where $\mathcal{F}(n) = \mathcal{F}(n-1) + \mathcal{F}(n-2)$ and $\mathcal{F}(1) = 1$ and $\mathcal{F}(2) = 1 \forall n > 2$ using strong induction.

Step 1: Show the “Basis”, which means simply, show for some finite values that the proposition holds for those values. For instance, let $a = 2$, show that say $P(a), P(a+1), \dots, P(5)$, that the proposition holds. These needn’t be exhaustive but it is a good step to do to get a feel for the problem itself. What would be a good a to pick for this problem?

Step 2: Show to inductive hypothesis, which means simply, assume $P(i)$ for $a \leq i \leq k$ where i is some index variable that runs between a (some arbitrary starting point) and k (the substitution for n , we must replace n with k to for tidiness). We do this to establish our finite collection of claims in a tidy way. As good computer scientists we want to keep our proofs and our code we write as tidy as possible. This may seem like a redundant step at times but it is arguably the most important. What is the inductive hypothesis for this problem?

Step 3: Show the inductive step, which means simply, prove $P(k+1)$. This involves introducing our inductive hypothesis into the $P(k+1)$. We assume that the inductive hypothesis is true. You must always introduce the inductive hypothesis into the inductive step. Finish the proof.

2. Calculate the following anti-derivatives from 1 to n with respect to x . (Because I am a nice person. Remember that $\log_2 x = \frac{\ln x}{\ln 2}$). I have run into all of these integrals when computing runtimes of algorithms, I know you didn't expect this to be so much math but please, please don't hate me. I added these because I care.

(a) $6 + 7x + 2x^2$

(b) $\frac{1}{x^2}$

(c) $\frac{1}{x}$

(d) 2^x (HINT: Use the fact that $\frac{d}{dx} A^x = \ln A * A^x$ where A is some finite real number)

(e) $\log_2 x$ (HINT: set $dv = 1 * dx$)

