Worksheet on Summations and Recurrences

Benjamin Cosman, Patrick Lin and Mahesh Viswanathan

Fall 2020

Takeaways from Lecture

- Closed forms for basic summations:
  - For an arithmetic sequence $a_1, \ldots, a_n$ with common difference d, the corresponding arithmetic series is $\sum_{i=1}^{n} a_i = \frac{n(a_1+a_n)}{2} = \frac{n(2a_i+(n-1)d)}{2}$.
    As a special case, $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$.
  - For a geometric sequence $a_1, \ldots, a_n$ with common ratio r, the corresonding geometric series is $\sum_{i=1}^{n} a_i = \frac{a_1(1-r^n)}{1-r}$.
    Equivalently, $\sum_{i=0}^{n-1} ar^i = \frac{a(1-r^n)}{1-r}$.
- Recurrences
  - A recurrence is a function defined in terms of itself.
  - Finding closed forms for recurrences can be done by unrolling:
    a) "Unroll" three or four times
    b) **Predict** the form obtained after unrolling k times
    c) Find the value of k that matches the base case(s)
    d) Substitute the base case(s) and simplify
    e) Sanity check on a few values, or, ideally, prove correctness by induction
  - For special recurrences of the form $T(n) = aT(\frac{n}{b}) + f(n)$ (a recursive calls and $f(n)$ extra work), we can use the recursion tree method. The root of the tree for $T(n)$ has value $f(n)$ and a children, each of which is the root of a (recursively constructed) recursion tree for $T(\frac{n}{b})$.
    a) Draw the first few layers of the recursion tree
    b) **Predict** the total extra work at each (internal) level
    c) Compute the number of internal levels as well as the number of leaves
    d) Add up the total work at each internal level, plus the work at the leaves
    e) Simplify sums and logarithms to obtain a closed form
    f) Sanity check on a few values, or, ideally, prove correctness by induction

Problem 1 (Remember how to do induction)**.** Prove that $\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$ via induction.

Problem 2 (Fun in the sum)**.** Find a closed form for the sum of the first n positive odd integers, i.e. $\sum_{i=1}^{n}(2i-1)$. [1]

[1] Hint: first split this sum up into the sum of some other sums.

Problem 3 (More Boolean Hypercubes)**.** Recall that the number of edges $EQ(n)$ in the boolean hypercube Q_n can be defined as

$$EQ(n) = \begin{cases} 0 & \text{if } n = 0 \\ 2EQ(n-1) + 2^{n-1} & \text{if } n > 0. \end{cases}$$

Compute a closed form for $EQ(n)$ via unrolling.

Problem 4 (Are you really a tree?). Consider the following recurrence, defined over non-negative powers of three (i.e., 3^k for $k \geq 0$):

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ T(\frac{n}{3}) + n^2 & \text{otherwise} \end{cases}$$

Compute a closed form for the recurrence via the recursion tree method.