Concrete Mathematics Notes

AUGPath

April 21, 2023

Chapter 1

Recursion Problems

1.1 The Tower of Hanoi

Problem setup. There are 3 pegs and some disks stacked in decreasing order.

- Rule. Move 1 disk at a time and never moving a large one to a smaller one

- Objective. Move all disk from one peg to another.

Technique 1. Look at small examples. Transfer with 1, 2 and 3 disks is somehow obvious. General patterns are easier to precieze while extreme cases are well understood.

Technique. Divide and conquer. We first transfer $n-1$ smallest to a different peg (T_{n-1} times in total), then move the largest, then move the smallest on the idea. So there are $2T_{n-1}+1$ times. So we have

$$T_n \leq 2T_{n-1}+1, \text{for } n > 0.$$

We have to move the biggest one, so when moving the biggest one, we have already moved the ones at the bottom. As we have to move it up. So we have

$$T_n \geq T_{n-1}+1, \text{for} n > 0.$$

Recurrence Relation.
Representation: Like

$$\begin{aligned} &T_n = 0 \\ &T_n = 2T_{n-1}+1, \text{for } n > 0 \end{aligned}$$

is called a recursion.

Solution: a closed form for this, for example, $T_n = 2^n - 1$. A closed form involves only addition, subtraction, multiplication and division and exponentiation in explicit ways.

Mathematical Induction. First we prove a statement when it has the smallest value(basis), then we prove it for $n \geq n_0$, assuming it has been proved in $[n_0, n-1]$(induction).

1.2 Lines on a plane

Problem setup. What's the maxim number L_n of regions defines by n lines on the plane?

Observing small cases.

Observation 1: Adding a new line seems to double the region.

Observation 2: adding the 3rd line, it can at most hit at least 3 regions. So the desired generalization is it splits k old regions if and only if it hits all previous regions. So the upper bound is

$$L_n \leq L_{n-1} + n, n > 0$$

This is a arithmetic series, so this the answer is $S_n = n\frac{n+1}{2}$.

Variation problem. With lines on the plane problem: Suppose instead of straight lines, using zigzag lines. What's the max. number?

Observation: If we extend all the lines up, it will be like the lines on the plane problem. Each line will lose n regions, so it will be

$$Z_n = L_{2n} - 2n, n \geq 0.$$

1.3 The Josephus problem

Problem setup. Start with n people $1 \sim n$, eliminate every second remaining person till only 1 survives. We shall determine the survival number.

Using proper notation. Denote $J(n)$ as the final survivor's number. Note then n is odd, we have $J(2n) = 2J_n - 1$. For after one round, the configuration is 2 times and minus 1. So the $J(2n+1) = 2J(n) + 1$.

Simplify. We note that this is closely related to power of 2's. After a few listing it can be shown that $J(2^m + l) = 2l+!$. We can prove it by induction.

It may be also helpful to see radix 2 representations. Suppose n's binary representation is $n = (b_m, b_{m-1}, \cdots, b_1, b_0)$. According to the previous ones, we have that $J(b_m, b_{m-1}, \cdots, b_1, b_0) = (b_{m-1}, \cdots, b_1, b_0, b_m)$. That is moving one bit cyclic shift left!

Guessing in a smarter way. Assume our recurrence had something like

$$\begin{aligned} f(1) &= \alpha \\ f(2n) &= 2f(n) + \beta \\ f(2n+1) &= 2f(n) + \gamma \end{aligned}$$

We can assume the general form like $f(n) = A(n)\alpha + B(n)\beta + C(n)\gamma$. After writing a few, we might investigate $A(n) = 2^m, B(n) = 2^m - 1 - l, C(n) = l$.

This method will work perfectly well when the function is linear.

1.4 Exercises

Here are some answers to the exercises. The answer might not be always correct.

W 1. In this problem, the set of the horses doesn't meet well ordering principle, that is no linear order for a set.

W 2. Note that splitting it in the first one and then other ones is not correct, for when it was set like this, it will cause big ones over small ones. This will lead to $T_n = 2 + T_{n-1} + 2 + T_{n-1} + 2$ – this is a wrong answer. The correct answer is using top $n-1$ as a group and one at last one, so hence swapping all Ts and 2s in the previous equation, hence getting $T_n = T_{n-1} + 2 + T_{n-1} + 2 + T_{n-1}$. Using techniques of inductions will get $3^n - 1$.

W 3. From the previous question we have totally $3^n - 1$ cases will be shown. We may find that no two configurations are the same, since each move at least one plate is in the different place. On the other hand, there are $3^n - 1$ cases according to rules of multiplication. So we are done.

W 4. No. Since it has reached the highest part.

W 5. We can derive the formula for circles. That is $T(n) = T(n-1) + 2n$. Calculating for $n = 4$, we have at most 14 areas. So this is impossible.

W 6. We can deduce it firstly to L_n problems. Subtracting surrounding unbounded regions, we have $B_n = L_n - 2n$.

W 7. This argument can only make sure the recurrence is true, without proving the basis one is correct.

H 8. We note that $Q_0 = \alpha$, $Q_1 = \beta$. Calculating for $Q_3 = \frac{1+\beta}{\alpha}$, and $Q_4 = \frac{\alpha+\beta+1}{\alpha\beta}$, and we find that $Q_4 = \frac{1+\alpha}{\beta}$, and

$$\begin{aligned} Q_5 &= \frac{\alpha\beta + \alpha^2 + \alpha}{\alpha + \beta + 1} = \alpha. \\ Q_6 &= \beta. \end{aligned}$$

So we end up in a loop. So the answer is:

$$Q_n = \begin{cases} \alpha & n \bmod 4 \text{ is } 0 \\ \beta & n \bmod 4 \text{ is } 1 \\ \frac{1}{\alpha\beta} + \frac{1}{\alpha} + \frac{1}{\beta} & n \bmod 4 \text{ is } 2 \\ \frac{1+\alpha}{\beta} & n \bmod 4 \text{ is } 3 \end{cases}.$$

HW 9. (a) We show that

$$x_1 x_2 \cdots x_{n-1} x_n \leq \left(\frac{x_1 + \cdots + x_n}{n}\right)^n$$
$$x_1 x_2 \cdots x_{n-1} \frac{x_1 + \cdots + x_{n-1}}{n-1} \leq \left(\frac{x_1 + \cdots + x_{n-1} + \frac{x_1 + \cdots + x_{n-1}}{n-1}}{n}\right)^n$$

Multiplying and making some ideas, we have

$$x_1 x_2 \cdots x_{n-1} \frac{x_1 + x_2 + \cdots + x_{n-1}}{n-1} \leq \frac{x_1 + x_2 + \cdots + x_{n-1}}{n-1}$$

and $P(n-1)$ is true now.

(b) We can set $x_1 = \frac{y_1 + \cdots + y_n}{n}$, and set $x_2 = \frac{y_{n+1} + \cdots + y_{2n}}{n}$.

(c) We may use the binary representation for each number, hence we can repeat the case over and over again until implying all the possible $P(n)$s.

HW 10.

Chapter 2

Manipulating sums

2.1 Basic Notations

Summation notation. We can use $\sum$ to denote add things together.

- $\sum_a$: sum over all items in a
- $\sum_{i=1}^n f(i)$: sum from $i = 1$ to $i = n$, substituting all is in the right $f(i)$.

Iverson's Bracket. We have the notation $[p]$: p is a proposition, and when p is true, the equation is evaluated 1, otherwise evaluated as 0.

Using Aversion's bracket to simplifying sums. For example we have

$$\sum_{p \text{ prime}, p \le N} p = \sum_p [p \text{ prime}][p \le N]p.$$

2.2 Manipulation of sums

Distributive law. $\sum_{k \in K} ca_k = c \sum_{k \in K} a_k$.

Associative law. $\sum_{k \in K}(a_k + b_k) = \sum_{k \in K} a_k + \sum_{k \in K} b_k$.

Commutative law. $\sum_{k \in K} a_k = \sum_{p(k) \in K} a_{p(k)}$.

This rule is for mainly substitution use. When the condition is not specified, the $p(k)$ should be a permutation of all integers.

Example. We have

$$\sum_{k \in K, k \texttt{ even}} a_k = \sum_{n \in K, n \texttt{ even}} a_n = \sum_{2k \in k}^{a_{2k}}$$

Example 1: Arithmetic's progression. We have $S = \sum_{0 \le k \le n}(a + bk)$. By commutative law, replace k by $n - k$, obtaining $S = \sum_{0 \le n-k \le n}(a + b(n - k)) = \sum_{0 \le k \le n}(a + bn - bk)$. These two can be added up to $2S = \sum_{0 \le k \le n}(2a + bn)$. Then this problem would be trivial.

Exclusion and Inclusive Principle with Sums. Here is a important rule for combining different set of indicies.

Suppose K and K' are any set of integers, then

$$\sum_{k\in K} a_k + \sum_{k\in K'} a_k = \sum_{k\in k\cap K'} a_k + \sum_{K\in K\cup K'} a_k.$$

In the case of Iverson's bracket, we have

$$[k\in K] + [k\in K'] = [k\in K\cap K'] + [K\in K\cup K']$$

Perturbation method. We first write $S_n = \sum_{0\le k\le n} a_k$, then we rewrite S_{n+1} in 2 ways:

$$\begin{aligned} S_n + a_{n+1} = \sum_{0\le k\le n+1} \qquad\qquad &= a_0 + \sum_{1\le k\le n+1} a_k \\ &= a_0 + \sum_{1\le k+1\le n+1} a_{k+1} \\ &= a_0 + \sum_{0\le k\le n} a_{k+1}. \end{aligned}$$

Now we can work on the last term trying to solving the closed form for it.

Example 2. Geometry progression. We have $S_n = \sum_{0\le k\le n} ax^k$. And the sum is needed. Using *perturbation* method we have

$$S_n + ax^{n+1} = ax^0 + \sum_{0\le k\le n} ax^{k+1}$$

Factoring out an x, we have

$$S_n + ax^{n+1} = a + xS_n$$

$$\sum_{k=0}^{n} ax^k = \frac{a - ax^{n+1}}{1-x}, \qquad \text{for } x \ne 1$$

Example 3. Arithmetic Geometric progression. We have the sequence

$$S_n = \sum_{0\le k\le n} k2^k$$

Using the perturbation technique, we have

$$S_n + (n+1)^{2n+1} = \sum_{0\le k\le n} (k+1)2^{k+1}$$

Rewrite RHS as sums, we have

$$S_n + (n+1)^{2n+1} = \sum_{0\le k\le n} k2^{k+1} + \sum_{0\le k\le n} 2^{k+1}$$

Hence $\sum_{0\le k\le n} k2^k = (n-1)2^{n+1} + 2$.

2.3 Multiple sums

Notations. Stacking multiple sums in a row helps to derivate more complex sums.

Example: $\sum_{i=1}^{3}\sum_{j=1}^{3} a_{ij} = a_{11} + a_{12} + a_{13} + \cdots + a_{33}$.

Using the simplified Iverson's bracket, getting another way of expressing multiple sums: $\sum_{1\le j,k\le 3} a_j b_k = \sum_{j,k} a_j b_k [1 \le j \le 3][1 \le k \le 3]$.

General distributive law. For distinct lower set index i, j, we have the following distributive law:

$$\sum_{j\in J,k\in K} a_j b_k = \big(\sum_{j\in J} a_j\big) + \big(\sum_{k\in K} b_k\big)$$

Another general form for this is $\sum_{j\in J}\sum_{k\in K(j)} a_{j,k} = \sum_{k\in K'}\sum_{j\in J'(k)} a_{j,k}$. Here, the sets should satisfy $[j \in J][k \in K(j)] = [k \in K'][j \in J'(k)]$.

Example 1. Consecutive integers. We may rewrite $[1 \le j \le n][j \le k \le n] = [1 \le j \le k \le n] = [1 \le k \le n][1 \le j \le k]$.

Example 2. Sum over an matrix. We try to find a simplified idea of the matrix

$$\begin{pmatrix} a_1a_1 & a_1a_2 & \cdots & a_1a_n \\ a_2a_1 & a_2a_2 & \cdots & a_2a_n \\ \vdots & \vdots & \ddots & \vdots \\ a_na_1 & a_na_2 & \cdots & a_na_n \end{pmatrix}$$

we shall find a sum $S_1 = \sum_{1\le j\le k\le n} a_j a_k$.

According to the associativity of the multiplication, we find that $a_j a_k = a_k a_j$. Hence we have $S_\Delta = \sum_{1\le j\le k\le n} a_j a_k = \sum_{1\le k\le j\le n} a_k a_j = \sum_{1\le j\le n} a_j a_k = S_\nabla$. According to the problem above, we have $[1 \le j \le k \le n] + [1 \le k \le j \le n] = [1 \le j, k \le n] + [1 \le j = k \le n]$. We have

$$2S_\nabla = S_\nabla + S_\Delta = \sum_{1\le j,k\le n} a_j a_k + \sum_{1\le j=k\le n} a_j a_k.$$

The first sum is $\left(\sum_{j=1}^{n} a_j\right)\left(\sum_{k-1}^{n} a_k\right) = \left(\sum_{k=1}^{n} a_k\right)^2$, and the second sum is that $\sum_{k-1}^{n} a_k^2$.

So we have the sum

$$\sum_{\nabla} = \sum_{1\le j\le k\le n} a_j a_k = \frac{1}{2}\left(\left(\sum_{k=1}^{n} a_k\right)^2 + \sum_{k=1}^{n} a_k^2\right).$$

The idea is like the aligning the data and subtracting the overlapped off. That's a more religious form.

Another double sum. We try to evaluate

$$S = \sum_{1\le j<k\le n} (a_k - a_j)(b_k - b_j).$$

We find that this sum satisfies summation:

$$S = \sum_{1\le j<k\le n} (a_k - a_j)(b_k - b_j) = \sum_{1\le j<k\le n} (a_j - a_k)(b_j - b_k).$$

We add S to itself, with equation

$$[1 \le j < k \le n] + [1 \le k < j \le n] = [1 \le j, k \le n] - [1 \le j = k \le n].$$

yields

$$2S = \sum_{1\le j,k\le n} (a_j - a_k)(b_j - b_k) - \sum_{1\le j=k\le n} (a_j - a_k)(b_j - b_k)$$

The second part is 0, so we have to evaluate the first sum. The first sum expand it and swapping the summation index we have the identity

$$2\sum_{1\le j,k\le n}^{n} a_k b_k + 2\sum_{1\le j,k\le n} a_j b_k$$

We have sums over j, k, extracting j out, we have

$$2n\sum_{1\le j\le n}^{n} a_k b_k + 2\left(\sum_{1\le j\le n} a_j\right)^2 \left(\sum_{i\le k\le n} b_k\right)^2$$

Chebyshev's monotonic inequalities . This is a special case for the last case as an example, that is

$$\left(\sum_{i=1}^{n} a_k\right)\left(\sum_{k=1}^{n} b_k\right) \le n\sum_{k=1}^{n} a_k b_k$$

if $a_1 \le a_2 \le \cdots \le a_n$ and $b_1 \le b_2 \le \cdots \le b_n$. and vice versa.

Changing the index as the single sums. In the index is changed in single sums, we have that

$$\sum_{k\in K} a_k = \sum_{p(k)\in k} a_k,$$

if $p(k)$ as the permutation of the integers.

We now generalize k by $f(j)$, where f is an arbitrary function:

$$f : j \to K$$

that takes an integer $j \in J$ into an integer $f(j) \in K$. In this case we have the replace formula:

$$\sum_{j\in J} a_{f(j)} = \sum_{k\in K} a_k \# f^{-}(k)$$

where the $\#f^{-}(k)$ stands for number of sets in the set $f^{-}(k) = \{j | f(j) = k\}$.

Proof. $\sum_{j\in J} a_{f(j)} = \sum_{j\in J, k\in K} a_k [f(j) = k] = \sum_{k\in K} a_k \sum_{j\in J} [f(j) = k]$. This yields the answer.

Example: A fraction sum. We wish to sum

$$S_n = \sum_{1\le j<k\le n} \frac{1}{k-j}.$$

An attempt shows that

$$\begin{aligned} S_n &= \sum_{j=1}^{n} \sum_{k=j}^{n} \frac{1}{k-j} \\ &= \sum_{1\le k\le n}^{1\le k-j<k} \text{Replacing } j \text{ by } k-i \end{aligned}$$