Epp chapter 5 (sequence, induction)

5.1 SEQUENCES

* CScientists often times have to deal with sequential data, that is, data that is organized linearly, so that each piece comes immediately before some other piece and immediately after some other piece (except for the very first and last pieces)
* Mathematicians call this type of structure a sequence.
* Data types used in sequences can be anything, but we start off by talking about sequences of numbers.
* Suppose I give you the sequence 2, 4, 6, 8, 10. Every element of a sequence is called a term. We often give names to the entire sequence like we give names to an entire list in Python.
* Also, just like in Python, we can use indices to talk about specific terms of the sequence. We write the indices as subscripts. The starting subscript can be anything you want, but often its either zero or one. Each further term increases the subscript by one.
  + a\_1 = 2, a\_2 = 4, a\_3 = 6.
* This is a finite sequence but we can also have infinite sequences.
  + When we’re talking about generic sequences, for a finite sequence we will usually write   
    a\_m, a\_m+1, … a\_n.
  + Infinite: am, am+1, …
* Would be really tedious if we had to define every sequence by writing it out, so we have some shortcut ways.
* Easiest way it to define a sequence with a formula, often based on the index or subscript.
  + a\_i = 2\*i for all ints i , 1 <= i <= 5.
  + B\_i = 2\*i for all ints i >= 1.
* Define a sequence so the terms are 1, 2, 4, 8, 16, …. [ex sol: c\_i = 2^i for all ints i, i>=0]
* Define a sequence so the terms are -1, 1, -1, 1, -1, 1, … [ex sol: d\_i = (-1)^i for all ints i, i>=1]
* Summation notation
  + Define sum of a\_i as i goes from m to n.
    - Infinite sums
  + Ex: sum of k^2 as k goes from 1 to 4.
  + Ex: convert from 1 + ½ + 1/3 + … + 1/n. = sum of 1/i as i goes from 1 to n.
  + Convert from 1/n + 2/(n+1) + 3/(n+2) + … + (n+1)/(2n)
    - Rewrite 2n as n+n
    - = (i+1)/(n+i) as i=0 to n.
* Separating off final term:
  + Ex: Rewrite sum(i=0, n, i) = sum(i=0, n-1, i)+n
  + Ex: rewrite sum(i=1, n+1, 1/n^2) = sum(i=1, n, 1/i^2) + 1/(n+1)^2
  + Ex: rewrite sum(i=0, k+1, i(i+1)/2) = sum(i=0, k, (k+1)(k+2)/2)
* Adding in an additional term.
  + Ex: rewrite sum(i=0, n, 2^i) + 2^n+1 = sum(i=0, n+1, 2^i)
  + Ex: rewrite sum(i=0, k, (i+1)/(i+2)) + (k+2)/(k+3) = [sum from i=0 to k+1]
* Product notation
  + Pi(i=m, n, a\_i) = a\_m \* a\_m+1 \*…\* a\_n-1 \* a\_n.
* Properties of summations/products
  + Sum ai + sum bi = sum (ai + bi)
  + Sum(constant \* ai) = constant\*sum(ai)
  + Prod ai + prod bi = prod(ai \* bi)
* Change of index:
  + Change sum of 1/i as i goes from 1 to n to start at 0.
  + Use different variable, say j. Set j = i-1. Then i = j+1.
    - Lower limit was i=1. Now is j=1-1 = 0.  
      upper limit was i=n. Now is j=n-1.
    - Term was 1/i. Now is 1/(j+1).
* Factorial notation.
  + n! = n\*(n-1)\*(n-2) … 3 \* 2 \* 1.
  + 0! = 1.
* **(SKIP)** Choose notation
  + N choose r = (n r)
  + If 0 <= r <= n, then (n r) = n!/(r!(n-r)!).
  + This formula arises because it’s the number of ways we can make a group of r objects from a bigger group of n objects.
    - Say you have a group of 4 students, called A,B,C,D. How many ways can you make a group of 2 of them.
    - AB, AC, AD, BC, BD, CD
    - 6 = 4!/(2! 2!) = (4 3 2 1)/(2 1 2 1) = (4 3 2 1) /4 = 3\*2 = 6.

5.2 INDUCTION I

* Define some new sets: Z+, Z>2, Z(nonneg)
* Today we’re going to introduce a new technique for proofs, called mathematical induction, or just induction.
* The idea behind mathematical induction is the following:
  + Let’s say I set up a row of dominoes on the table.
  + Suppose I tell you two things:
    - All the dominoes are spaced so that if domino k falls, then domino k+1 falls.
    - I push over domino 1.
  + What can we conclude about what happens next? (all the dominoes fall over).
  + What if it’s an infinite sequence of dominoes (they still all will fall over, eventually, given enough time.)
* Mathematical induction says suppose I have a predicate P, that is defined on integers.
  + I want to prove P is true for all integers >= a. (a is usually specified in the problem as a specific number)
  + If I prove two things: 1. P is true for a.
  + And 2. For all k in Z(>=a), P(k) -> P(k+1)
  + Then we have proved for all n in Z>=a, P(n) is true.
* Method for doing induction proofs:
  + Identify what P(n) is, and what a is.
  + Two separate proofs:
    - Basis Step: Show P(1) is true. [book calls this basis step]
    - Inductive Step: Show for all k in Z>=a, P(k) -> P(k+1)
      * This is done by generic particular:
      * Assume k is an arbitrary integer in Z>=a, and P(k) is true. Prove P(k+1).
        + Name the Inductive hypothesis
* First proof: prove for all n in Z^>=1, sum of first n pos integers = n(n+1)/2.
  + Rewrite using sum: SUM(I = 1 to n of i) = n(n+1)/2
    - Note that induction in general, cannot be used to find formulas for sums of sequences, only to PROVE the formula is true once we’ve found it.
  + P(n) is the entire statement with the equals sign, not just one side.
    - Note that P(n) is not a number. Like sin(n) is a number. P(n) is a proposition that is either true or false for n.
  + **Base Case**: Prove P(1) is true.
    - Rewrite P(1) out formally.
    - How do we prove an equation is true?
      * Transform left side into right side.
      * Transform both sides together into the same thing.
    - 1 =? (1(1+1))/2
    - 1 =? 2/2
    - 1 = 1
    - BASE CASE DONE
    - ***Difference between proving an equation is true, and solving an equation for a variable.***
  + **Inductive case**: prove forall k in Z>=a P(k) -> P(k+1)
    - Off to the side, write out P(k) and P(k+1)
    - Suppose k is an arbitrary integer that is >= 1.
    - Assume P(k) is true.
    - Sum(I, i=1 to k) = k(k+1)/2
    - Prove P(k): sum(I, i=1 to k+1) = (k+1)((k+1)+1)/2.
      * Notice how I used the word “prove,” indicating this is something that we don’t know is true yet. Therefore, we cannot use this statement in our proof. We want to use a piece of it to know where to start, but we cannot use the entire equation in the proof. This equation will only appear at the end, once we have derived it from other things.
    - How do we prove this?
      * Prove an equation is true by transforming the left into the right or right into the left, or left and right into the same third thing.
    - During this part of the proof, we must use the inductive hypothesis somewhere in here. Often the trick to each induction problem is figuring out how to use the inductive hypothesis.
      * Inductive hypothesis is usually an equation. There is always some part of an induction proof where you substitute in LHS of the IH for the RHS.
      * 2 techniques: start from the inductive hypothesis and try to transform one side of it into something that relates to what we want to prove.
      * Start with one side of what we want to prove, and transform it into something that looks like one side of the inductive hypothesis.
      * If you’re proving a summation, I like to start with the summation side of what we’re trying to prove and transform it into the RHS.
    - Sum(I, i=1 to k+1). How can we rewrite this as something that looks like something from the IH.
    - Sum(I, i=1 to k+1) = sum(I, i=1 to k+1) + sum(I, i=k+1 to k+1)
    - = sum(I, i=1 to k+1) + (k+1)
    - = k(k+1)/2 + (k+1) **BY IH**
    - = k(k+1)/2 + 2(k+1)/2
    - = [k(k+1) + 2(k+1)]/2
    - = (k+1)(k+2)/2 = (k+1)(k+1+1)/2
    - DONE
* **SKIP** 2nd proof: for all r in reals neq 1, for all n in Z>= 0, sum(r^I, i=0 to n) = (r^(n+1)-1)/(r-1).
  + P(n): sum = formula
  + But P(n) has an r in it, so we will actually begin the proof with GP on r.
  + Base case:
  + Inductive case:
    - IH =
    - IS: sum(r^I, i=0, k+1) = break up sum
    - = sum(r^I, i=0, k) + r^(k+1)
    - = (r^(k+1)-1)/(r-1) + r^(k+1)
* **DO**: Alternate version of second proof, do it for r = 2.
  + Prove sum(2^I, i=0 to n) = 2^(n+1)-1.

INDUCTION II (5.3)

Notes to start 3/14/16:

* Remind students how to succeed in class
  + Keep up with reading
  + You must understand every proof we do. Suggest being able to write them out completely on your own.
* Difference between HAVE and WANT. Keep separate.
* Review: proof for all n in Z >= a P(n)
  + Steps: define P(n)
  + Basis step: Prove P(a)
  + Inductive step: Prove for all k in Z >=a P(k) -> P(k+1)
    - Do this by assuming P(k) is true [IH]
    - And proving P(k+1) is true
* List of “tricks”
  + Proving summation formula: re-write sum from i=a to k+1 as sum from a to k + [k+1 term]
* Proving a divisibility property.
* Prove for all n >= 0 3 | 4^n – 1
  + HAVE: 3p = (4^k) – 1.
  + WANT: 3(int) = [4^(k+1)] – 1
  + Two ways to do this.
    - FIRST WAY: Start with RHS of thing to prove.
    - 4^[k+1] = 4\*4^k – 1 = (3+1)4^k – 1 = 3\*4^k + 4^k – 1
    - Now use IH on 4^k – 1.
    - SECOND WAY: Start with RHS of thing to prove.
    - 4^[k+1] = 4\*4^k – 1.
      * PAUSE: Work with IH. 3p = 4^k – 1. Therefore 3p+1=4^k.
      * Substitute 3p+1 for 4^k.
    - 4^[k+1] = 4\*4^k – 1 = 4(3p+1) – 1 = 12p + 4 – 1 = 12p + 3.
  + Trick is rewriting 4 as 3+1.
  + Often the trick with divisibility proofs is re-writing some coefficient using a sum or difference
* Inequalities:
  + With a true inequality, you can
    - add/subtract same thing from both sides
    - multiply by a pos number
    - multiply by a neg # and switch the sign
* Prove for all n >= 3. 2n + 1 < 2^n
  + Have: 2k+1 < 2^k
  + Want: 2k+3 < 2^(k+1).
  + Middleman technique.
    - We are trying to prove a < c. We’re going to find a b and show a < b and b < c.
    - First side will be proved using IH.
    - Second side
* Define seq: a\_1 = 2. a\_k = 5(a\_k-1).
  + Prove for all n>=0, a\_n = 2\*5^(n-1).
* GROUP WORK:

Prove For all n>=0, Sum of (2i+1) from i=0 to n = (n+1)^2

* + Prove for all n >=0, 3^k+2k is odd.

5.4: STRONG INDUCTION

* Some situations where induction is not powerful enough.
  + Alternate method of induction called strong induction.
  + Normal induction: I push over the first domino, and if domino k falls over, then domino k+1 falls over.
  + Strong induction: I push over a few dominoes at the beginning, and if all the dominoes 1, 2, 3, up to domino k falls over, then domino k+1 falls over.
    - Imagine the dominoes are getting bigger and bigger. And so each successive domino requires the force of all the previous dominoes to get it to fall over.
* Def’n of strong induction:
  + Suppose I want to prove for all n >= a, P(a) is true.
  + If I prove two things
  + Base Case: P(a), P(a+1)…P(b) are all true.
  + Inductive case: for all k in Z>= b, if for all integers I from a through k, P(i), then P(k+1).
* Ex: define a0 = 0, a1 = 4, ak = 6a\_(i-1) – 5a\_(i-2)
  + Prove for all n >= 0, an = 5^n – 1.
  + Try to do this with regular in
* Ex: for all n >= 2, n is divisible by a prime number.
  + Equiv: for all n >= 2, there exists a prime p such that p | n.
  + Base case
  + Inductive case: Show there exists a prime p such that p | k+1
    - Case 1: k+1 is prime. Done.
    - Case 2: k+1 is not prime. Then it’s composite. There exist ints r, s > 1 and < k+1 such that k+1 = rs.
    - By IH, r and s are divisible by primes.
      * So there is a prime p and an integer k such that pk = r.
    - K+1 = rs = pkr.
      * Because kr is an integer by closure of Z, p | k+1.

ASIDE: A complete binary tree of height h has 2^(h+1) – 1 nodes.

Next day

G1=3 g2=5

Gk = 3gi-1 - 2g i-2

Prove gn = 2^n+1 for all >=1

Use strong induction to show F\_n < 2^n for all n>=1.

Suppose we define a sequence as follows: a1 = 1; a2 = 3; and for all integers i ≥ 3, ai = ai−2 + 2ai−1. Prove that every term in the sequence is odd.

Suppose we define a sequence as follows: b0 = 2; b1 = 7; and for all integers i ≥ 2, bi = 3bi−1 − 2bi−2. Prove ∀n ∈ Z ≥0 ∃q ∈ Z bn = 5q + 2. (Alt version: 5 | bn – 2).

IND PRACTICE:

Weak induction:

For all n>=0, Sum of (2i) from i=0 to n = (n)(n+1)

For all n>=0, Sum of (2i-1) from i=1 to n = n^2

For all n>=0, Sum of (2i-1) from i=0 to n = n^2-1

For all n>=0, Sum of (2i+1) from i=1 to n = n(n+2)

For all n>=0, Sum of (2i+1) from i=0 to n = (n+1)^2

For all n>=0, Sum of (3i^2-i) from i=0 to n = (n^2)(n+1)

For all n>=0, Sum of (3i^2+i) from i=0 to n = (n)(n+1)^2

For all n>=0, Sum of (6i^2-4i) from i=0 to n = (n)(n+1)(2n-1)

For all n>=0, Sum of (6i^2+4i) from i=0 to n = (n)(n+1)(2n+3)

For all n>=0, Sum of (1/(i+2) – 1/(i+1)) from i=0 to n 1/(n+2) – 1

For all n>=1, Sum of (1/(i+1) – 1/i) from i=1 to n 1/(n+1) – 1

For all n>=1, Sum of (1/i – 1/(i+1)) from i=1 to n n/(n+1)

For all n>=1, Sum of (1/(i+1) – 1/(i+2)) from i=1 to n 2n/(2n+4)

For all n>=1, Sum of (1/(i+3) – 1/(i+2)) from i=1 to n n/(3(n-3)) = 1/(n+3)-1/3

For all n>=0, Sum of (i \* i!) from i=0 to n = (n+1)! – 1

For all n>=1, prod of (2i) from i=1 to n = 2^n \* n!

For all n>=1, prod of (i+1)\*i from i=1 to n = (n+1) (n!)^2 = (n+1)!\*n!

For all n>=2, prod of (i^2-i) from i=2 to n = (n!)^2 / n = n! \* (n-1)!

For all n>=2, prod of (i^3-i^2) from i=2 to n = (n!)^3 / n

For all n>=1, prod of (i+1)/i from i=1 to n = n+1

For all n>=1, prod of (i+1)/i from i=2 to n = (n+1)/2

For all n>=1, prod of i/(i+1) from i=1 to n = 1/(n+1)

For all n>=1, prod of i/(i+1) from i=2 to n = 2/(n+1)

For all n>=1, prod of (i+1)/i^2 from i=1 to n = (n+1)/(n!)

For all n>=1, prod of (i+1)/i^2 from i=2 to n = (n+1)/(2 \* n!)

For all n>=1, prod of i/(i^2-1) from i=2 to n = 2n/[ (n+1)! ]

For all n>=1, prod of (i+1)/(i^2-1) from i=2 to n = 1/(n-1)!

For all n>=1, prod of (i+1)/(i-1) from i=2 to n = n(n+1)/2

[Pick an integer a] For all n>=0, a^n – 1 is divisible by (a-1).

[Pick an integer a] For all n>=0 a^n can be written as (a-1)k + 1 for some int k.

Example: 3^n can be written as 2k+1

For all n >= 0, 2^n <= (n+2)!

A\_1 = 1

A\_n = a\_(n-1) + n^3.

Prove for all n>= 1, each a\_n is a perfect square

OR

Prove for all n>= 1, a\_n = [ n(n+1)/2 ]^2

A\_1 = 1

A\_2 = 2

A\_n = n \* (n-1) \* A\_(n-1)

Prove a\_n = n! \* (n-1)! [ NOT VERIFIED ]

Strong:

A\_1 = 1

A\_2 = 3

A\_3 = 5

A\_n = A\_(n-1) + A(n-2) + A(n-3)

Prove every term is odd.