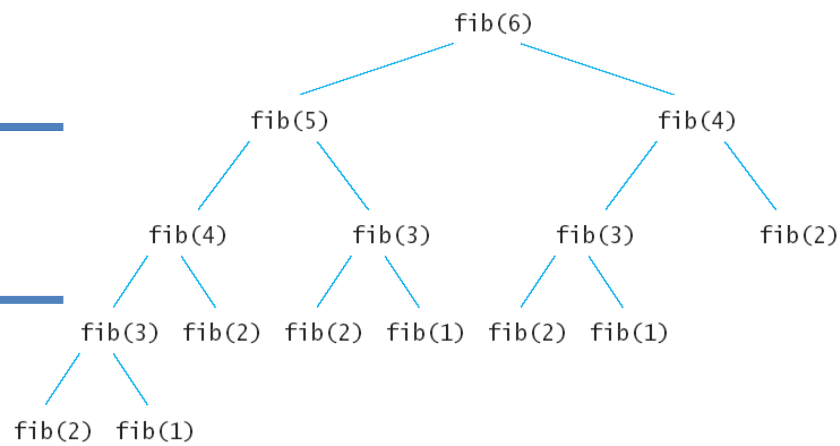
COMP 2121
DISCRETE MATHEMATICS

Lecture 12



Sequences

Definition:	*Sequence is an ordered list of numbers* (*or objects*).

- **Sequences defined explicitly**

Example 1. Given a formula, find the terms of $a_k = \dfrac{k}{k+1}, k \geq 1$ and $b_i = \dfrac{i-1}{i}, \quad i \geq 2$

Example 2. Given the first k terms, find the formula:

1, 1/4, 1/9, 1/16, 1/25, 1/36, …

Example 3. List the terms of alternating sequence $a_k = (-1)^k, \quad k \geq 0$ and find an explicit formula to describe the following sequence: 1, - 1/4, 1/9, - 1/16, 1/25, -1/36, …

Sequences defined recursively

Consider the following sequence of numbers:

$$0, 1, 1, 2, 3, 5, 8, 13, 21, \ldots$$

It is the case that each element of the sequence (other than the first two elements) is the sum of the previous two elements. For example, $2 = 1 + 1$; $21 = 8 + 13$ and so on. This sequence is called *Fibonacci sequence*, and can recursively be defined as:

$$a_n = a_{n-1} + a_{n-2},\ \ n>2 \text{ where } a_1=0 \text{ and } a_2=1.$$

We have learned that $n! = n\cdot(n-1)\cdot\ldots\cdot3\cdot2\cdot1$. Factorial can be expressed recursively too. Let $a_n = n!$. Also, $a_{n-1} = (n-1)!$ and it follows that $a_n = n\cdot a_{n-1}$. However, we still need to define initial condition (i.e. the first element) and that is $a_1=1$.

Example 4. Given that, $a_n=5a_{n-1}$, $n>1$ and $a_1=7$ find a_{100} .

Example 5. Give a recursive definition for the sequence defined as $a_n = 4n - 1$, $n \geq 1$.

- **Summations**

We use the following notation:

$$\sum_{k=m}^{n} a_k = a_m + a_{m+1} + a_{m+2} + \ldots + a_{n-1} + a_n$$

Note that *k* is a variable of summation or dummy variable. You could use a *j* instead:

$$\sum_{j=m}^{n} a_j = a_m + a_{m+1} + a_{m+2} + \ldots + a_{n-1} + a_n$$

We can sometimes simplify summation by changing the variable and the limits of the range, for example:

$$\sum_{k=0}^{6} \frac{1}{k+1} = \sum_{j=1}^{7} \frac{1}{j}$$

Example 6. Transform the following summation by making a change of variable: $j = i - 1$

$$\sum_{i=1}^{n-1} \frac{i}{(n-i)^2}$$

Example 7. Think of the variable of summation as the variable that controls a loop in a program

Version A:

```
n = 0;
for (i = 0; i < 10; i++)
  n = n + i;
```

Version B

```
n = 0;
for (j = 1; j <= 10; j++)
  n = n + (j - 1);
```

Version C

```
n = 0;
for (k = 17; k >7; k--)
  n = n + (17 - k);
```

Does each of the above code examples result in the same value of *n*? Which code example would be preferable?

- **Multiplying terms of a sequence**

Product Notation:

$$\prod_{k=m}^{n} a_k = a_m \cdot a_{m+1} \cdot a_{m+2} \cdot \cdots \cdot a_n$$

Example 8. Given that $a_k = \dfrac{k}{k+1}$, $k \geq 1$, find a closed formula $\prod_{k=3}^{n} a_k$.

- **Properties of Summations and Products**

If $a_m, a_{m+1}, a_{m+2}, \ldots$ and $b_m, b_{m+1}, b_{m+2}, \ldots$ are sequences of (real) numbers and c is any (real) number, then the following equations hold for any integer $n \geq m$

1.	$\sum_{k=m}^{n} a_k + \sum_{k=m}^{n} b_k = \sum_{k=m}^{n} (a_k + b_k)$
2.	$c \cdot \sum_{k=m}^{n} a_k = \sum_{k=m}^{n} c \cdot a_k$
3.	$\left(\prod_{k=m}^{n} a_k\right) \cdot \left(\prod_{k=m}^{n} b_k\right) = \prod_{k=m}^{n} a_k \cdot b_k$

Example 9. Given that $\sum_{k=1}^{99} k^2 = 328,350$ calculate $\sum_{k=1}^{100} (2 + k^2)$

Example 10. Given the formula $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

a) compute 25 + 26 + 27 + 28 +…+ 92 + 93.

b) compute the sum $\sum_{i=3}^{2n+1} (4i - 3)$

Example 11. The following pseudocode sorts array A, consisting of n elements, in increasing order. How many time units does it take to execute the pseudocode?

****You should note that the two for loops are nested loops.*

```
(1)     for i = 0 to n-2 do {
(2)         min = i
(3)         for j = (i + 1) to n-1 do {
(4)             if (A[j] < A[min]) {
(5)                 min = j
(6)             }
(7)         }
(8)         swap A[i] and A[min]
(9)     }
```

Assume the following:

- ✓ It takes insignificant amount of time to execute any statement other than statements in lines (4) and (8);
- ✓ It takes 2 time units for a single execution of the statement in line (4): if (A[j] < A[min]);
- ✓ It takes 12 time units for a single execution of the statement in line (8): swap A[i] and A[min].

Use formula $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$.