Analysis of Algorithms - CSCI 323
Lecture #2 - February 10, 2016

Notes by: Katherine Sandoval

Homework

P1. Define upper bound & lower bound

$$n^2 < 1^2 + 2^2 + 3^2 + \cdots + n^2 < n^3$$

$$an^3 + bn^2 + cn + d$$

$n=0 \quad a0^3 + b0^2 + c0 + d = d = 0$

$n=1 \quad a1^3 + b1^2 + c1 + \quad = a+b+c = 1$ 1 term in this case

$n=2 \quad a2^3 + b2^2 + c2^1 = 8a+4b+2c = 5$

$n=3 \quad a3^3 + b3^2 + c3^1 = 27a+9b+3c = 14$

① $a+b+c=1$

② $8a+4b+2c=5$

③ $27a+9b+3c=14 \Rightarrow 24a+6b=11$

② − ① $6a+2b=3$

③ − ① $24a+6b=11$

$6a+0b=2$

$6a=2$

$\boxed{a=\frac{1}{3}}$

$6(\frac{1}{3})+2b=3$

$2b=3-2=1 \quad \boxed{b=\frac{1}{2}}$

$\frac{1}{3}+\frac{1}{2}+c=1$

$c=1-\frac{1}{2}-\frac{1}{3} = \boxed{\frac{1}{6}=c}$

$$\frac{n^3}{3}+\frac{n^2}{2}+\frac{n}{6} \Rightarrow \boxed{\frac{n(n+1)(2n+1)}{6}}$$

P2) Base case

$n = 0$ works.

Inductive Hypothesis

Assume $1^2 + 2^2 + 3^2 + \cdots + k^2 = \frac{k(k+1)(2k+1)}{6}$

Prove $\underbrace{1^2 + 2^2 + \cdots + k^2}_{} + (k+1)^2 = \frac{(k+1)(k+2)(2k+3)}{6}$

Using inductive Hypothesis to simplify $\frac{k(k+1)(2k+1)}{6} + \frac{6(k+1)^2}{6}$

$$\frac{(k+1)}{6}\left[k(2k+1) + 6(k+1)\right]$$

$$= \frac{k+1}{6}\left(2k^2 + \overbrace{k + 6k}^{7k} + 6\right)$$

$$= \frac{k+1}{6}\left((k+2)(2k+3)\right)$$

P3) $a + (a+d) + (a+2d) + \cdots + (a+nd)$

Add all a's: $(n+1)a + d\underbrace{[1+2+\cdots+n]}_{\frac{n(n+1)}{2}}$

$$(n+1)\left[a + \frac{dn}{2}\right]$$

P4)

$$g(n) = c + cr + cr^2 + \dots + cr^n$$
$$rg(n) = cr + cr^2 + cr^3 + \dots + cr^{n+1}$$

$$rg(n) - g(n) = cr^{n+1} - c$$

$$g(n) = \frac{cr^{n+1} - c}{r-1} = \boxed{c\left[\frac{r^{n+1}-1}{r-1}\right]}$$

P5) 0 1 1 2

$$P(n) = an^3 + bn^2 + cn + d$$
$$P(0) = 0 = d$$
$$P(1) = 1 = a + b + c$$
$$P(2) = 1 = 8a + 4b + 2c$$
$$P(3) = 2 = 27a + 9b + 3c$$
$$P(4) = 6$$
$$a = \frac{1}{3} \qquad b = \frac{-3}{2} \qquad c = \frac{13}{6}$$

$$\boxed{P(n) = \frac{n^3}{3} - \frac{3n^2}{2} + \frac{13n}{6}}$$

P6) Rank functions ① Brake them into groups: Polynomial exponential

$n^2 \log n \log n \leftarrow (n \log n)^2 \rightarrow n^2 \log^2 n$

$\log \log n$
$n^{\sqrt{2}}$
2^n
$(\log n)^2$
$n!$
2^{2^n}
n^e
e^n
$2^{\ln n}$
$(\sqrt{2})^n$

$\log_2(2^{\ln n}) \rightarrow \ln n \rightarrow n^{\ln 2}$

Loganthmic	Polynomial	Exponential	
① $\log\log n$	⑤ $(n \log n)^2$	⑧ 2^n	⑩ $n!$
② $(\log n)^2$	④ $n^{\sqrt{2}}$	⑨ e^n	⑪ 2^{2^n}
	⑥ n^e	⑦ $\sqrt{2}^n$	
	③ $n^{\ln 2} = 2^{\ln n}$		

For Solving $2^{\ln(n)}$

$$\log_c a^{\log_c b} = \log_c b^{\log_c a}$$

$$(\log_c b)\log_c a = \log_c a \log_c b$$

$$2^{\ln n} = n^{\ln 2} \rightarrow \text{btw } 0 \text{ \& } 1$$

$$\Rightarrow \cancel{n^2} \log n \log n \ ? \ \cancel{n^2} \cdot n^{.73}$$

$$\sqrt{1{,}000{,}000} = 1000 \qquad \log_{10} 1{,}000{,}000 = 6$$

$$\Rightarrow \log_2 2^{2^n} = 2^n$$

P7) $f(n) = o(g(n))$

$$h(n) = \frac{g(n)}{f(n)} \qquad \log\left(\frac{g(n)}{f(n)}\right) \times f(n)$$

$$\log\log\left(\frac{g(n)}{f(n)}\right) \times f(n)$$

They are Asymtoptily smaller than previous one but still bigger than $f(n)$

Less than $g(n)$ but bigger than $f(n)$.

- Time Complexity
- Space Complexity

{ Best case
average case — All possible sets of inputs & divide # of cases
worst case

- Best case - relatively intersting ?
No. Can hope for best but prepare for worst.

- Worst case - Very important to know upper bound

- Average case vs. Worst case → Depends on the worst case

- Put all extra checking to determine for best case situation. By trying to impress by best case, might actually increase the worst & average case.

- Average case - Take all escenarios analasy & divide by the number of escenarios.

- We'll derive all worst cases.

⇒ $f(0) = 0$
$f(1) = 1$
$f(n) = f(n-1) + f(n-2)$ $\quad 2^n$

memory table — bottom up
Recursive one

Closed - form

Characteristic Equation

$$f_n - f_{n-1} - f_{n-2} = 0$$
$$x^n - x^{n-1} - x^{n-2} = 0$$
$$x^{n-2}(x^2 - x - 1) = 0$$

Divide through:
$x^2 - x - 1 = 0$ — quadratic equation

$$ax^2 + bx + c = 0$$
$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Quadratic equation

$ax^2+bx+c=0$ 2 solutions.

$$X = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$$

$$= \frac{1 \pm \sqrt{(-1)^2-4(1)(-1)}}{2(1)}$$

$$= \frac{1 \pm \sqrt{5}}{2}$$

Linear homogeneous equ.

When we have 2 solutions:

$$P\left(\frac{1+\sqrt{5}}{2}\right)^n + q\left(\frac{1-\sqrt{5}}{2}\right)^n$$ Rule

Base case:

$f(0)=0$ $n=0$ $P\left(\frac{1+\sqrt{5}}{2}\right)^0 + q\left(\frac{1-\sqrt{5}}{2}\right)^0 = 0$

$$P + q = 0 \qquad P = -q$$

$f(1)=1$ $P\left(\frac{1+\sqrt{5}}{2}\right)^1 + q\left(\frac{1-\sqrt{5}}{2}\right)^1 = 1$

$$\frac{P+P\sqrt{5}}{2} + \frac{(-P)\cancel{q} - \overset{(+P)}{\cancel{q}}\sqrt{5}}{2} = 1 \qquad \frac{P+P\sqrt{5}}{2} \frac{-P+\sqrt{5}P}{2}$$

$$\frac{2P\sqrt{5}}{2} = 1$$

$$\frac{\cancel{2}P\sqrt{5}}{\cancel{2}} = 1$$

$$P = \frac{1}{\sqrt{5}}$$

$$q = \frac{-1}{\sqrt{5}}$$

Dominant

$$f(n) = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$$

Fib

produces Fib ints.

$\frac{1+\sqrt{5}}{2}$

1.6^n

as $n \to \infty$

0 (zero)

$(1.6^x)^{\frac{n}{x}}$

What x do I use to make the base 10?

$$\log_{1.6} 10 = \frac{\log_{10} 10}{\log_{10} 1.6}$$

$x \approx 4.9$

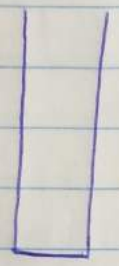
$$\boxed{f(n) \approx 10^{\frac{n}{5}}}$$

To approximate fib.

Review of Data Structures

ADT - Abstract datatype
you can't really have one.
Interface: contract between implementer & promoter.

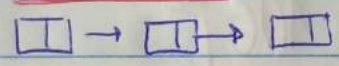
- **Stack** - abstract datatype.
  What features does it have? what it offers.



Take from top

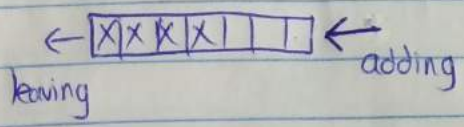
PUSH - Add element to the stack to the top
POP - Remove element.
top() - Tells what's at the top
isEmpty() -
numberOfElements() -
Capacity - How much can it hold before it is full.

implement an stack : Array better because do push/pop in the same place.
Linked List



- **Queue** (line)
  enqueue Put something
  dequeue
  isEmpty
  isFull - if capacity
  Peek - What's at the beginning of the queue

Array



- Shift everything down as one leave.
  n to move over 1

- Circllar implementation - Mantain pointers at the beginning & end.

- **List**

  insert (key, position) where to insert it.
  Delete (key)
  Delete (position) //overloaded method.
  iteration - iterate through list
  Sort
  isEmpty
  isFull
  numberOfElements
  Search (key) - might be part of delete
  get Keys
  Swap / re-arrange
  next / prev - Find out next position & previous one.

How do you implement a list? - Array: • size (allocate space)
• Delete in the middle. (moving everything around)

Trees

node (vertex) Children — root

- Smaller things to the left.
- Bigger things to the right.

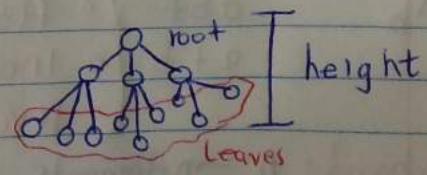
- Binary trees node can have upto 2 children.
- K-ary trees No max # of children.

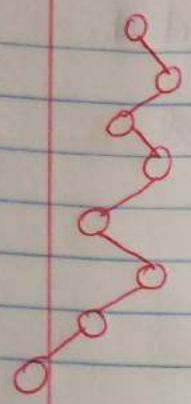
Binary tree tradeoff vs n-ary tree

Dictionary - Place to look something up/insert / retrieve.

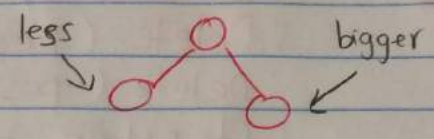
The more children I have -

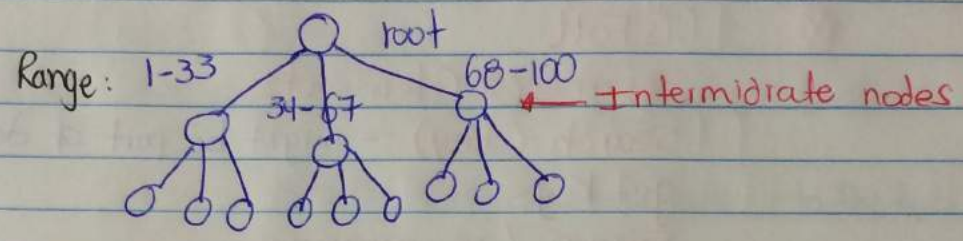
n objects in dictionary: $\log_K n$





Binary tree





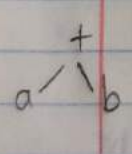
→ All children
Right Child / Left Child
Root
Insert / delete Nodes

Rotation - idea is to reorganize, so it won't end up with an snare-like tree.
- Move things around.

Traversal - Getting all data in certain order
- Depth-First Search (dfs) root - left - root - right
- breadth-first Search (bfs)
  Level order.

• Use a queue to implement it. Add in left to right level. Children at the back.

Order n



+ a b
a b +
a + b

- Preorder root, preorder(left), preorder (right)
- Postorder postorder(left), postorder(right), root
- Inorder inorder (left), root, inorder (right)
- reverse Order reverseOrder (right), root, reverseOrder (left)

Lowerbound: n → n elements
Upperbound:

Worst Case : looking at each one individually: $n-1$

Good time for a tree operation : $\log n$ or less

- How to mantain the tree (balance) to be able to do all operations on $\log n$.

Heap (Priority Queue) → most important key at top
Shape as a binary tree.

Smallest at top and children will be larger.
- You want a heap when you're interested in smallest thing at each point.
- Bubble Up / Bubble Down

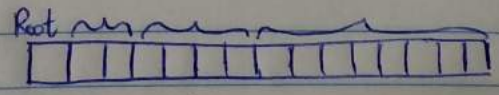
- delete - min : Deletion of the smallest element.
- When sorting to have smallest thing at the top

- Delete - Max : bulgar.
- 0-1 napsack either take item or leave it
Value per weight

Implementation of trees:

- LinkedList

- Array : issues
- waste of space
If it is not a Full balanced tree.
→ Make sure the kind of data we are working with.



2^L where L is the level

ch 4,5,6