Exam: first page posted ... 8 Qs / 60 ...
office hour(s) — next week + 16th

ps4 questions
today + tomorrow

CSC165 fall 2019

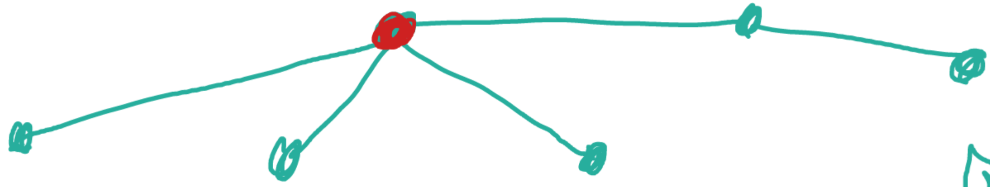
rooted trees / what's next

Danny Heap
csc165-2019-09@cs.toronto.edu
BA4270 (behind elevators)

Web page:
http://www.teach.cs.toronto.edu/~heap/165/F19/

Using Course notes: trees

distinguish a root



add notions of distance, hierarchy/direction to trees by

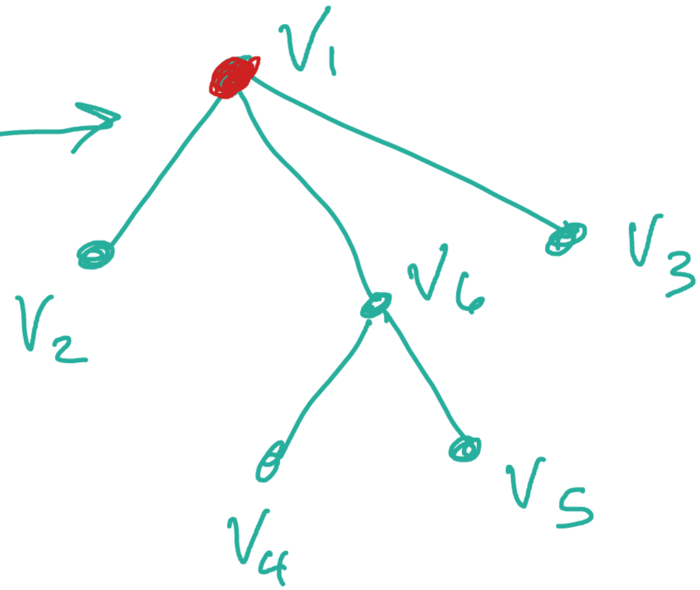
rooted tree: a tree with

- exactly one vertex labelled (distinguished) as root, if the tree has at least one vertex

- OR no vertices (a convenience for proofs and algorithms) picture below.

jargon

arity 3.



- parent → v_4, v_5

  v_6

- child → v_6

  v_4, v_5

- ancestor parent OR ancestor of parent
- descendant child OR descendant of child
- arity (~~branching~~ max factor)
- height, denote as $height(G)$

leaf - 0 children

interior - 1 or more children

easy-ish facts

→ prove as exercises!

- every rooted tree with $n \geq 2$ vertices has height at least 2
- some rooted tree with $n \geq 2$ vertices has height exactly 2
- every rooted tree with n vertices has height no more than n
- some rooted tree with n vertices has height exactly n

binary rooted trees

$P(h)$

maximum degree 3 $\equiv$ maximum of 2 children

$\forall h \in \mathbb{N}, \forall G = (V, E)\,(G \text{ rooted, binary tree } \wedge \mathit{height}(G) \leq h) \Rightarrow |V| \leq 2^h - 1$

Prove by induction on h.

base case, $P(0)$: only such tree is empty tree ($\to$), which has $0 = 1 - 1 = 2^0 - 1$ vertices. This verifies $P(0)$.

inductive step: Let $h \in \mathbb{N}$ and assume $P(h)$ $\leftarrow$ IH. Let G be a binary tree of height $\leq h+1$. Then G_L and G_R are binary trees rooted at G's left, right child respectively. They each have height $\leq$ 1 less than G (or shorter), hence height $\leq h$. So, by IH, G_L has $\leq 2^h - 1$ vertices and G_R has $\leq 2^h - 1$ vertices. So

$$G \text{ has } \leq \underbrace{1}_{\text{root}} + 2^h - 1 + 2^h - 1$$

$$= 2^{h+1} - 1 \text{ vertices} \quad \blacksquare$$

later topics...

- prove correctness — csc236 prerequistes $\Rightarrow$ post condition
- analyze recursive runtime — recurrence relations — Ω, Θ

csc463

- computability — impossible to compute.

csc373

- intractability — looooong time : 2^n
- public-key cryptography

problem with keys... e.g. Vigenere cipher

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

key: thewalrusandthecarpenter

cleartext: ifsevenmaidswithsevenmopssweptforhalfayear

bmw

ifsevenmaidswithsevenmopssweptforhalfayear

thewalrusandthecarpenterthewalrusandthecar

how do you securely exchange keys?

public/private

share public key with the world → everybody

keep private key secret → to myself.

allows:

authentication — encrypt with private, decrypt with public

encryption — encrypt with public, decrypt with private.

RSA initials inventors

need: text→integer, integer→text reversible padding scheme

≥ 1000 bits!

1. randomly choose **large** primes p and q
2. $n = pq$ (key length is n in bits...)
3. $L = (p-1)(q-1)$
4. choose $1 < e < L$ so that $\gcd(e, L) = 1$
5. compute inverse, $d \equiv e^{-1} \pmod{L}$, i.e. $de \equiv 1 \pmod{L}$ (notes Example 2.19 works for **co-prime**!)

publish: e, n

keep private d, p, q, L.

$m =$ text → integer(message)

encrypt: $c \equiv m^e \pmod{n}$

decrypt: message = integer → text$(c^d \pmod{n})$

it works... how?

Use results from this course... mostly

$m^p \equiv m \pmod{p}$

$p \mid m^p - m$

$\Rightarrow p \mid m(m^{p-1} - 1)$

$p \mid m^{p-1} - 1$

- $c^d \equiv m^{ed} \pmod{n}$
- $n = pq$, and $ed \equiv 1 \pmod{(p-1)(q-1)}$, i.e. $ed = 1 + k(p-1)(q-1)$
- $m^{ed} \equiv m \times m^{(p-1)(q-1)k} \pmod{p} \equiv m \times 1^{(q-1)k} \pmod{p}$ (problem set #3, Q1(c) almost...) $\equiv m \pmod{p}$
- also $m^{ed} \equiv m \pmod{q}$
- (problem set #2, Q2(a)): $m^{ed} \equiv m \pmod{pq} \equiv m \pmod{n}$.