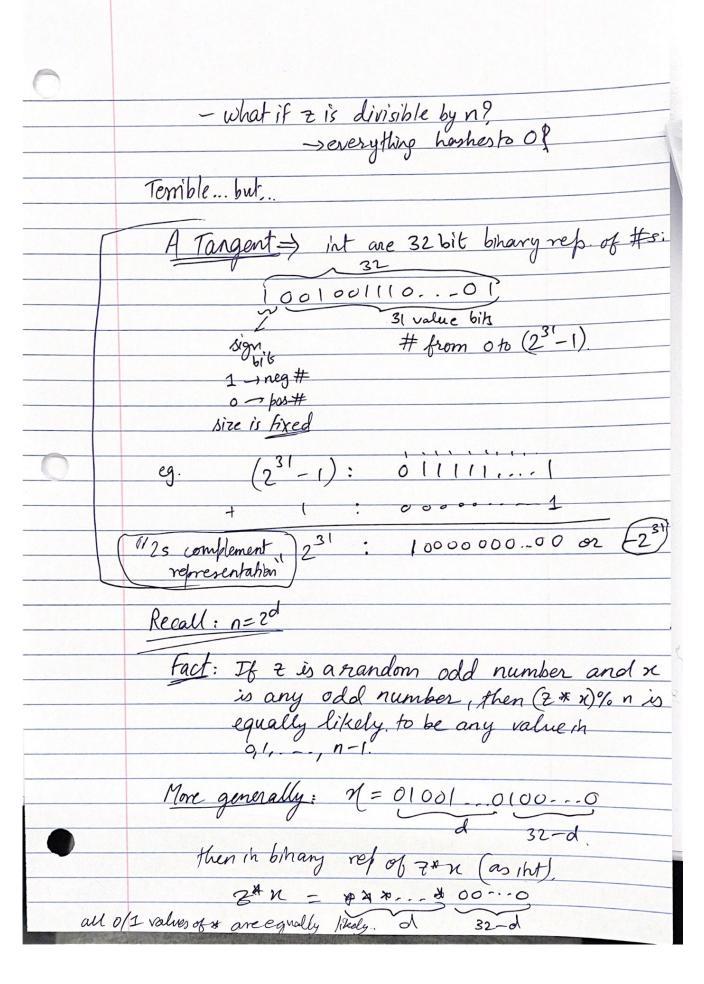
(7)	Lecture 19: Hash Code & Implementations
	generating ideas for hash tables:  -> how to define h(x)?
	-> how to define h(x)?
	Terminology: a hashfunction h is a rule for
	Terminology: a hashfunction h is a rule for assigning (numerical) values to object instances.
	- h(x) = hash code of x or hash value
	Goal: Fix a raile for generating hash codes where values appear random  - collisions are unlikely.  (x and y collide if h(x) = h(y) but
	- collisions are unlikely.
	n = 4)
0	
	In Java: Object class has a method
	(hash Code ()): for any object, obj, calling obj. hash (ode () returns an
	obj. hash (ode () returns an
	int that is a hash code for obje
	Guarantees:
	(1) if obj.hashCode() involved multiple
	same object gets same value.
	(2) if o.equals (p), then
	o. hash(ode() = p. hash(ode())
	but reverse is not necessarily true
	Non-Guarantee:
	"as much as reasonably possible", hash Codel) should avoid callisions
	should avoid callisions

Conclusion: Defining hash Code () for a class is an art form. > should NOT assume that default implementation is great ... "trust no one". Two issues: forward implementing hashTable 1) N. hashCode() - negative value (Index out of bounds 2) hash Gode may have patterns (not random) and lead to poor performance Goal: Use hashCode() method to generate indices for a hash table of size n. Ideal: If o.hashCode() + p.hashCode() then, p(h(0) = h(b)) = 1 (= 1) Simple approach: n = hash (ode (ob)) (some int) determine h(x) = |x| % n there between 0 and n-1make positive Problem (Doesn't satisfy ideal: if n=0, y=n  $\Rightarrow h(x) = 0, h(y) = 0$ Diff. way of harry zatrandom (int) -define has h(n) = (Z\*x) of on Better or worse than before?



Consequence: if n + y, and z is to an odd number chosen at random, then the first d-bits of z\*x and z\*y are equals with probability of 1 = 1) (randomnus over choice of z Suggests: given 2 = 065. hash (ode () define h(n) = first of bits of zxx. In code: >>> = bit shift operator. y = b3, b30 b29 ..... bo (6175 of y) first d bits of y is (y>>> (32-d)