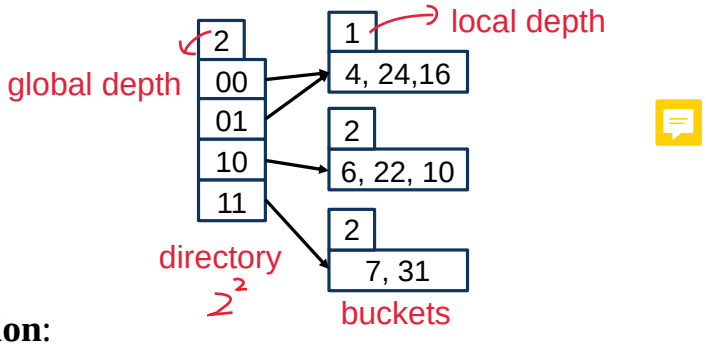
INFS4205/7205 Advanced Techniques for High Dimensional Data

Tutorial 2: Spatial Data Organization 1

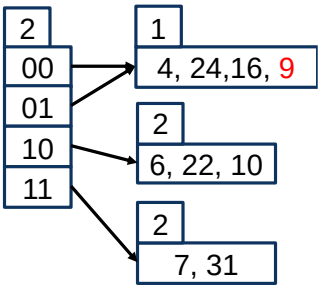
Semester 1, 2021

Question 1: Extendible hashing avoids overflow by doubling the directory size. Given the initial hashing structure below and the bucket size is 3, show how it changes when adding 9, 20, and 26. Suppose we use the suffix instead of the prefix as the hash function.

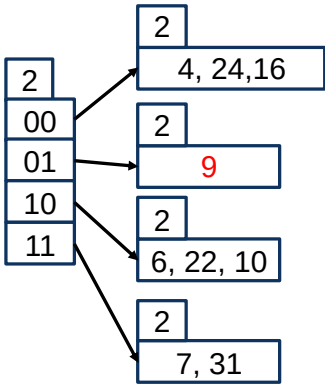


Example Solution:

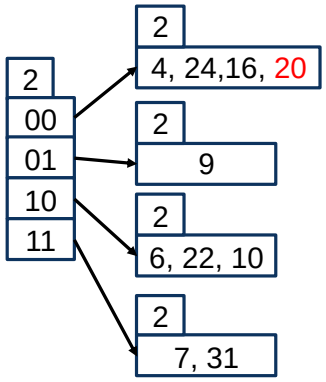
(1) Add 9. 9 mod 4 is 1, so add 9 to bucket 01, which goes to bucket (4,24,16,9), and it overflows.



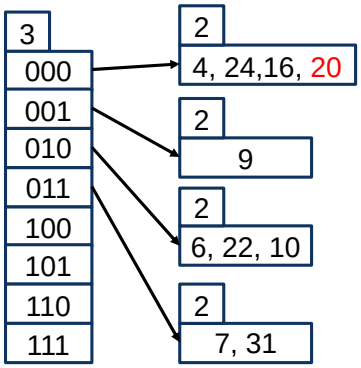
(2) The local depth is 1, which is smaller than the global depth 2, so we split this bucket without double the directory.



(3) Add 20. 20 mod 4 is 0, so we add 20 to bucket 00.

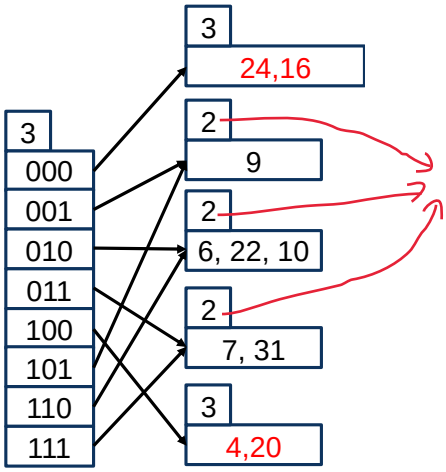


(4) Bucket 00 has four values, and the local depth is the same as the global depth, so we double the directory size.



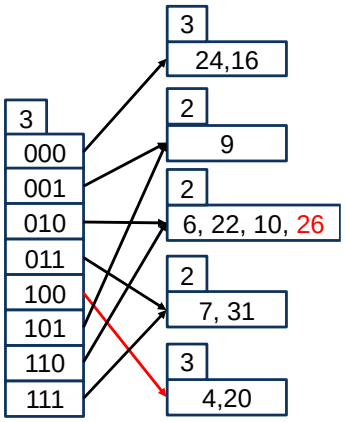
mod 8:
4 -> 100
24 -> 000
16 -> 000
20 -> 100

(5) Split the overflowed bucket 000 and we get two new bucket 000 and 100, whose local depths are all 3. 24 and 16 mod 8 are 0, so they stay in 000. 4 and 20 mod 8 are 4, so they go to say 100.

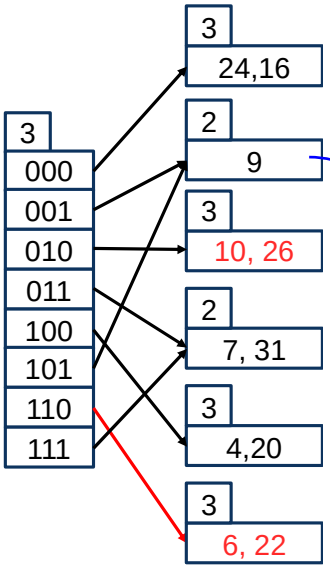


local depth =2, so last 2 bits are used as index

(6) Add 26. 26 mod 8 is 2, so add 26 to bucket 010, and it overflows.



(7) Because the bucket 010's local depth is 2, which contains the data base on mod 4 rather 8, we split it by upgrading to depth 3. 10 and 26 mod 8 are 2, so they stay in bucket 010. 6 and 22 mod 8 are 6, so move them to bucket 110.



Think after class:
1. what if we insert 17, 25, 33 to the bucket of 9?
2. what if use prefix as index?

Question 2: Linear hashing is a dynamic data structure which can grow or shrink one bucket at a time. Suppose $m = 2$, the load factor is $\frac{|item|}{|bucket\ capacity|} > 75\%$, and the bucket capacity is 2. The original hash table is shown below. Please show how the hash table changes if we insert the following items: 10, 15, 19, 22, 18

number of initial buckets

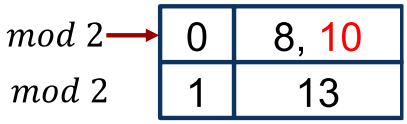
$mod\ 2$ →	0	8
$mod\ 2$	1	13

$h_0(k) = k \bmod (2^0 \times m)$

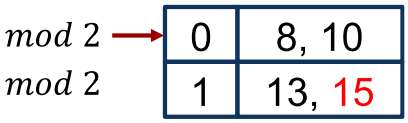
$h_i(k) = k \bmod (2^i \times m)$

Example Solution:

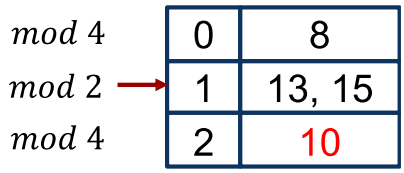
(1) Insert 10. Load factor $\frac{3}{2\times2} = 75\%$



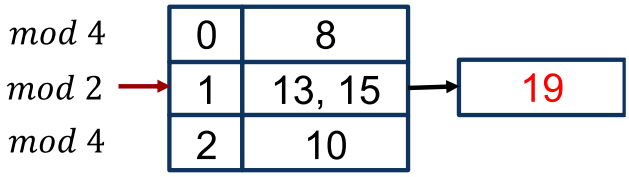
(2) Insert 15. Load factor $\frac{4}{2\times2} = 100\%$>75%



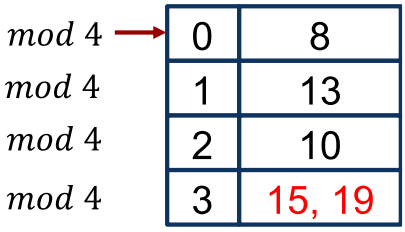
(3) Split 0 to 0 and 2, upgrade their m to 4, arrow moves down to 1.
Load factor = $\frac{4}{3\times2} = 67\%$



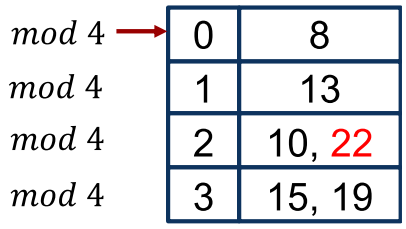
(4) Insert 19, 1 grows, Load factor $\frac{5}{3\times2} = 83\%$



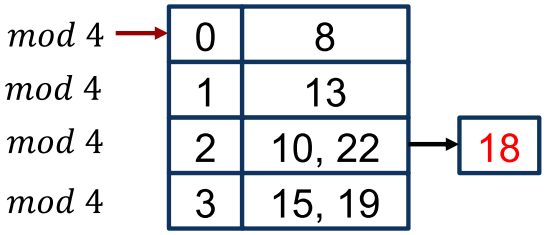
(5) Split 1 to 1 and 3, upgrade their m to 4, arrow returns to 0. Load factor = $\frac{5}{4\times2} = 62.5\%$



(6) Insert 22, load factor = $\frac{6}{4\times2} = 75\%$



(7) Insert 18, grows at the end of 2. load factor = $\frac{7}{4\times2} = 87.5\%$



(8) Split 0 to 0 and 4, upgrade their m to 8, arrow moves to 1. Load factor = $\frac{7}{5\times2} = 70\%$ No guarantee that a split relieves the overloaded bucket