

# Linear Hash

Friday, March 27, 2020 2:16 PM

Linear hashing is a very popular hashing method.

Let's for simplicity assume that we store integer values, and that the base hash function  $h = h_0$  is  $h(x) = x \bmod m$ . Initially, we have just  $M=m$  and use only  $h_0$ . We may have the following hash table ( $m=4$ ):

0: 4, 12, 20, 32  
1: 9, 13  
2: 6, 10, 30  
3: 7

Let's assume that the lists we store have maximum length 4, so the cell with  $h_0(x) = 0$  is already full. If we now insert 24, it should go into the full cell with hash value 0. As this is not possible we use now  $h_1$  for those  $x$  that are divisible by 4. We increase  $M$  to 5 and use in addition  $h_1(x) = x \bmod 2m$ . Whenever  $h_1(x) < M$ , we use  $h_1$ , otherwise we stay with  $h_0$ . This gives the following hash table:

0: 32, 24  
1: 9, 13  
2: 6, 10, 30  
3: 7  
4: 4, 12, 20

If we add 2 and 34 (both with  $h_0(x) = 2$ ), we must increase  $M$  to 7. This gives the hash table:

0: 32, 24  
1: 9, 13  
2: 10, 2, 34  
3: 7  
4: 4, 12, 20  
5:  
6: 6, 30

Actually, 13 should then move to the cell 5, but this can be handled in a lazy way, i.e. only do it when necessary.