

$$h(k) = k \% m$$

$$m = 13$$

	0	1	2	3	4	5	6	7	8	9	10	11	12
T	18	13	15	16	31	5	26						
next	1	4	-1	-1	6	0	-1	-1	-1	-1	-1	-1	-1

first empty: $\emptyset, 1, 4, 6, 7$

$$h(5) = 5 \% 13 = 5$$

$$h(8) = 18 \% 13 \Rightarrow 5$$

$$h(16) = 16 \% 13 = 3$$

Representation

HashTable:

h TFunction

m integer // capacity

firstEmpty integer

T TElem[]

next integer[]

- Subalg insert (ht, e)

if ht.firstEmpty = ht.yon then

 ⊙ resize and rehash

 p ← ht.h(e)

if ht.T[p] = NULLTELEM then

 ht.T[p] ← e

 ht.next[p] ← -1

 if p = ht.firstEmpty then

 ChangeFirstEmpty(ht)

else

 ht.T[ht.firstEmpty] ← e

 ht.next[ht.firstEmpty] ← -1

while ht.next[p] ≠ -1 ex:

 p ← ht.next[p]

ht.next[p] ← ht.firstEmpty

ChangeFirstEmpty(ht)

⊙

Not a good fit for hashtable:

- any sorted container
- stack
- queue
- list

Open addressing - Linear probing:

$$h(k, i)$$

↑
probe number

$$k: \langle h(k, 0), h(k, 1), \dots, h(k, m-1) \rangle$$

probe sequence

→ should be a permutation