

موضوع: Map و جدول درهم سازی

این جلسه می خواهیم راجع به MAP یا ADT Dictionary صحبت کنیم.

Dictionary (Map): داده ساختاری است که تعدادی دوتایی به صورت  $(key, value)$  نگه می دارد.  
 ↓ مقدار      ↓ کلید

عملیات:  $insert(key, value)$ : دوتایی را داخل داده ساختار درج می کند.

$search(key)$ : کلید داده شده را در دیتابیس جستجو و در صورت یافتن

مقدار value آن را برمی گرداند.

$delete(key)$ : کلید داده شده را جستجو و دوتایی مربوط به آن را پاک می کند.

\* کلید ها یکتا.

مثال:  $(94703217, 20)$ ,  $(98204432, 16)$ ,  $(99302741, 10)$

$16 \leftarrow search(98204432)$

$\leftarrow search(92307741)$

$delete(99302741)$

کاربرد: دیتابیس - کامپایلر - دفترچه تلفن.

key	۱۲۷۸	۲۴۳۲	...		
value	ضلع	ضلع	...		

۱ - آرایه نامرتب:

نوشته: کلیدها تکراری نیستند.  $insert: O(1)$

$Search: O(n)$  تعداد دوتایی ها

$delete: O(n)$  میانگین و بیشترین

key				
value				

۲ - آرایه مرتب بر حسب کلید.

$insert: O(n)$

$Search: O(\log n)$

$delete: O(n)$

۳ - استفاده از درج متوازن (AVL و B+)



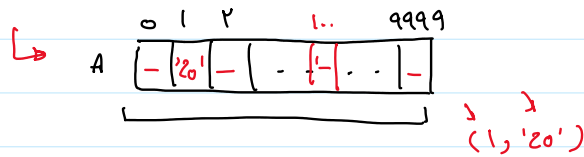
$insert: O(\log n)$

$delete: O(\log n)$

delete :  $O(\log n)$  ✓  
search :  $O(\log n)$  ✓

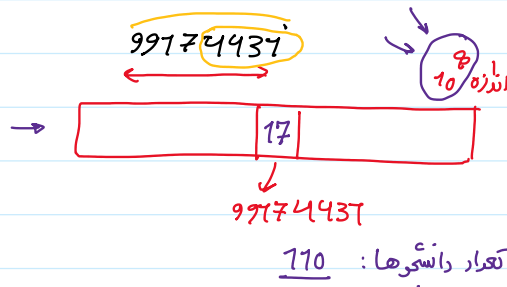
\* آیا می شود به عمل کرد؟  
در حالت متوسط

مثال ساده : فرض کنید می داریم تمام کلیدها، اعدادی در بازه  $\{0, 1, 2, \dots, 9999\}$  است.

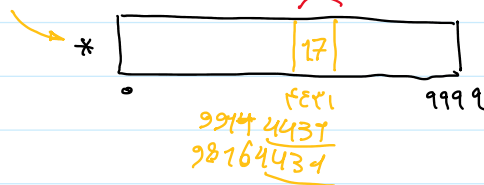


\* زمان درج :  $O(1)$   
\* زمان جستجو :  $O(1)$       Search (100)  
\* زمان حذف :  $O(1)$

• Direct Addressing  
مسئله : حافظه . تعداد عناصر بسیار کمتر از بازه کلیدها باشد



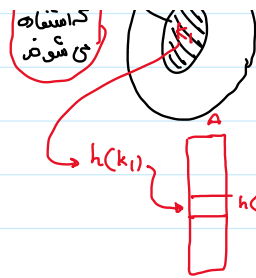
راه حل پیشنهادی : صرفاً ۱۷ رقم کوچک را در نظر بگیریم :



جدول درهم سازی ( Hash Table ) :



مجموعه کلید :  $U$  اندازه  $U$  خیلی بزرگ است  
ورودی شامل  $n$  دوتایی [مثال کلاس ماه  $n=10$ ]  
هر جدول درهم سازی ۱۷ خانه دارد :

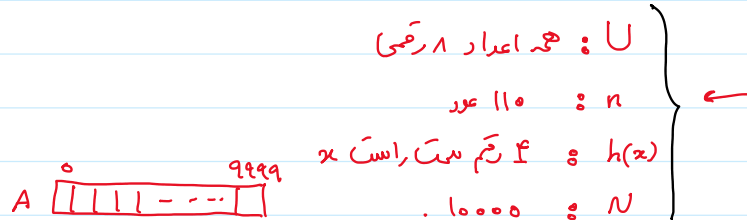


هر جدول درهم سازی ۲ المان دارد:

\* تابع درهم سازی:  $h: U \rightarrow [0, \dots, N-1]$  و  $U$ ،  $h(x) = \frac{x \times N}{A}$

$N$ : اندازه آرایه هش  
 $n$ : آرایه  $A [0, \dots, N-1]$

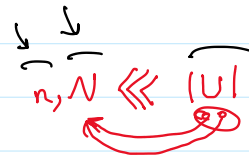
ایده تابع درهم سازی: برای دج کلید  $x$ ، مقدار  $h(x)$  را در  $A[h(x)]$  ذخیره کنیم.



$n \ll N$

تصادم: دو کلید مختلف به یک خانه map شوند:

$h(x) = h(y)$  و  $x \neq y$



دقت کنید چون اندازه  $U$  خیلی بیشتر از  $N$  است، تصادم غیر قابل اجتناب است.

چالش اصلی: رفع تصادم؟

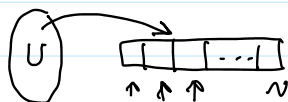
اولی: پستگیری بهتر از درما: تابع هش خود را خوب انتخاب کنید.

مثلا در بحث شماره دانشجویی ها ۱ رقم سمت چپ خوب نیست!

مثلا اگر  $h(x) = (ax + b) \% N$  و  $gcd(a, N) = 1$  و  $a \times N \neq 0$

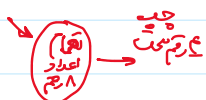
نوشت:  $h(x) = \lfloor N(\frac{ax + b}{N}) \rfloor$  و  $0 \leq a < N$  و  $0 \leq b < N$

پسینها



تابع هش ایده آل:

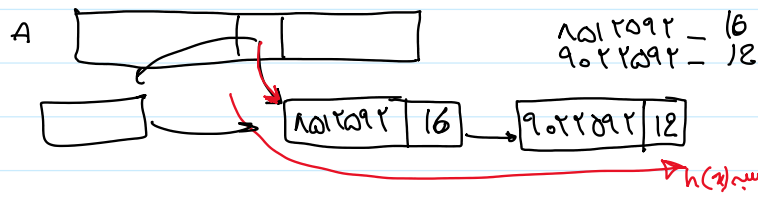
\* اگر یک کلید تصادفی از  $N$  انتخاب کنیم با احتمال بسیار کم به هر کدام از خانه ها map شود.



زنج تصادفی:

ع-نم-نوع  
↑  
value

1- روش زنجیه بندی  
separate chaining



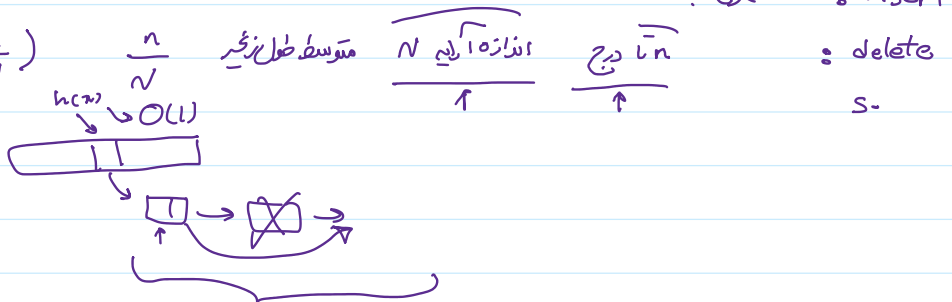
$h(x)$  : راصاب ی کنیم  
 $insert(n, val)$  : یک عدد جدید شامل  $(n, val)$  به اول آن اضافه می کنیم  
 $O(1)$   
 $O(1)$   
 $O(1)$   
 $h(x)$  : Search (x) : و تا انتهای زنجیه جستجو می کنیم  
 $O(n)$   
 $h(x)$  : delete (x)  
 $O(n)$

\* حالت متوسط : فرض  $n$  کلید درج شده هر کدام به صورت یکنواخت و تصادفی از  $U$  انتخاب شده اند و تابع درهم سازی ایده آل

load factor :  $\alpha$

ضریب بارگذاری

$$O(1 + \frac{n}{N})$$



Search : موفق :  $O(1 + \frac{\alpha}{4})$

ناموفق :  $O(1 + \alpha)$