

# Hash tablice

ALGORITMI I STRUKTURE PODATAKA

# Hash tabele

- ▶ Predstavljaju generalizacijo polja (nizova)
- ▶ Dinamička struktura za čuvanje objekata indeksiranih ključevima
- ▶ Insert
- ▶ Search
- ▶ Delete

# Primer

$$h(x) = x \bmod 13$$

►  $h(x,i) = (h(x) + i) \bmod 13$

► 18, 41, 22, 44, 59, 32, 31, 73

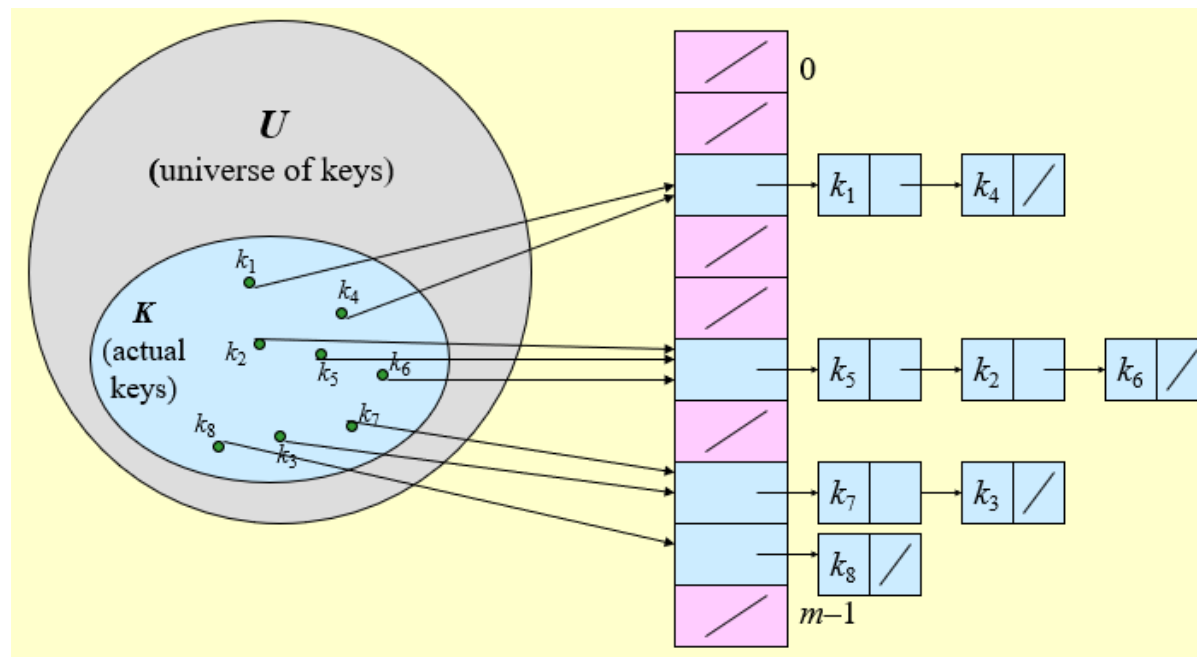
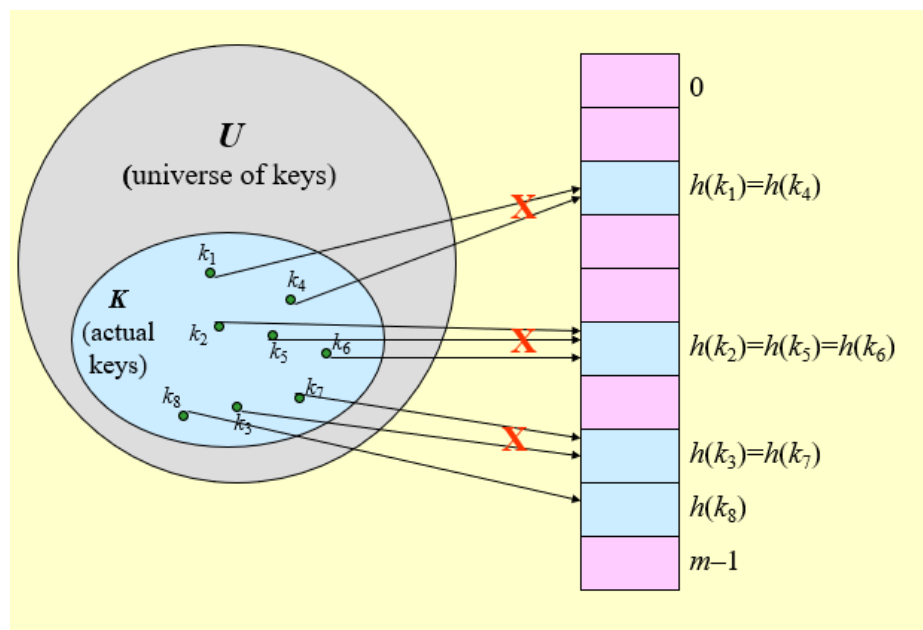
-	-	-	-	-	-	-	-	-	-	-	-	-
<u>0</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>	<u>12</u>



-	-	41	-	-	18	44	59	32	22	31	73	-
<u>0</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>	<u>12</u>

# Kolizija

- ▶ Više ključeva ukazuje na istu lokaciju
- ▶ Dva rešenja



# Pseudo kod za operaciju Insert

Hash-Insert( $T, k$ )

1.  $i \leftarrow 0$
2. **repeat**  $j \leftarrow h(k, i)$
3.       **if**  $T[j] = \text{NIL}$
4.       **then**  $T[j] \leftarrow k$
5.       **return**  $j$
6.       **else**  $i \leftarrow i + 1$
7. **until**  $i = m$
8. **error** "hash table overflow"

# Pseudo kod za operaciju Search

Hash-Search ( $T, k$ )

1.  $i \leftarrow 0$
2. **repeat**  $j \leftarrow h(k, i)$
3.       **if**  $T[j] = k$
4.       **then return**  $j$
5.        $i \leftarrow i + 1$
6. **until**  $T[j] = \text{NIL}$  **or**  $i = m$
7. **return** NIL