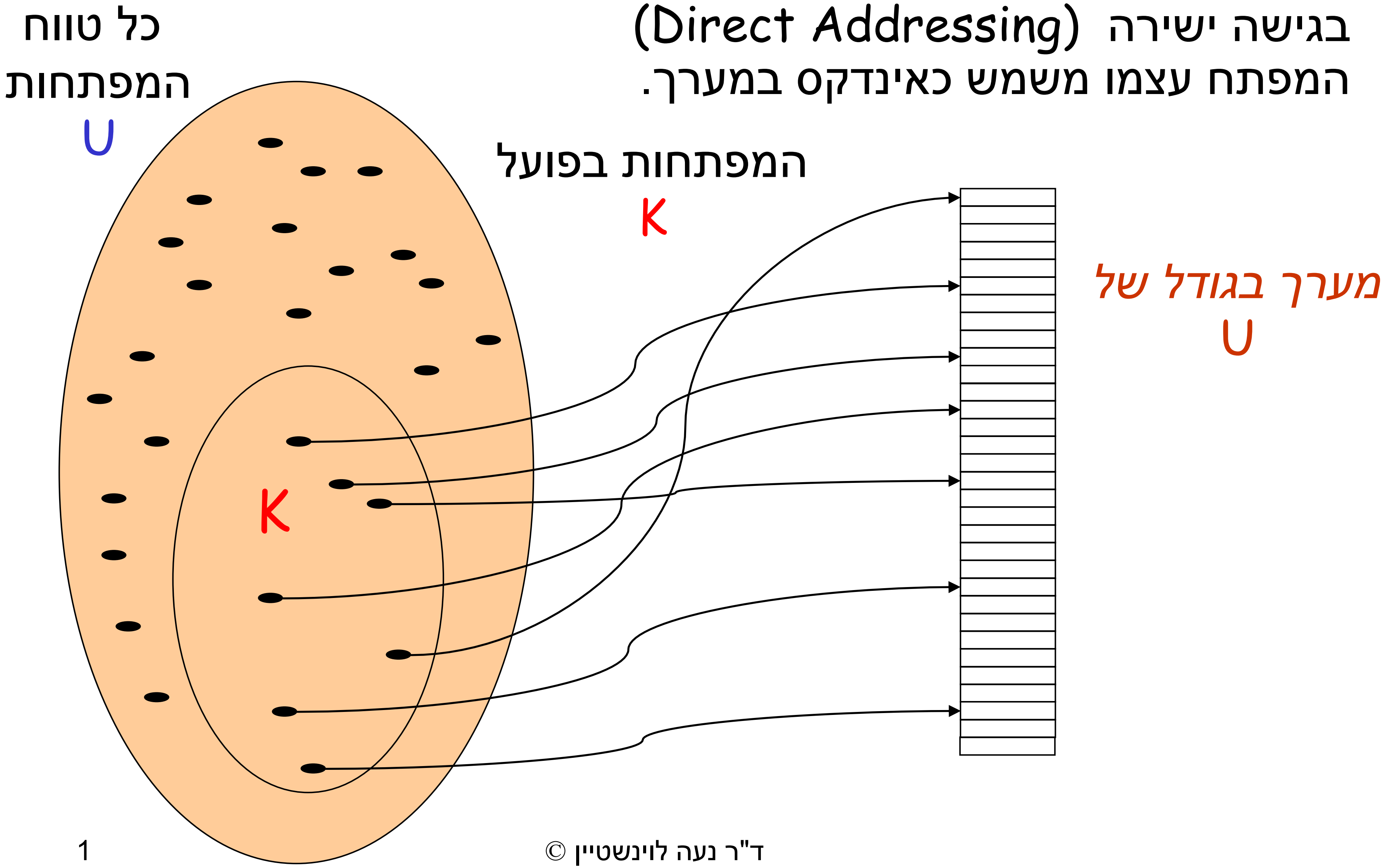
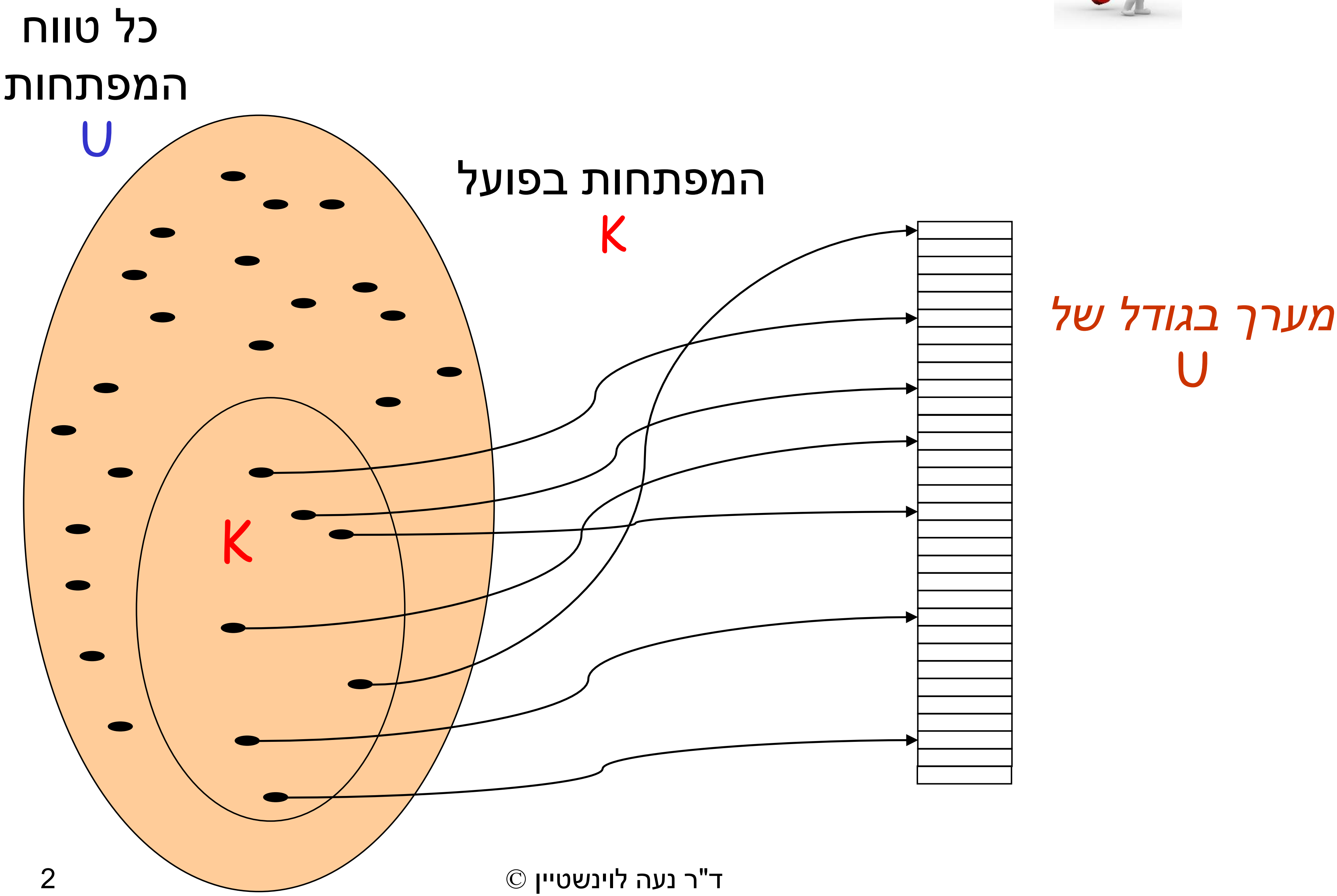
טבלת מיעון ישיר

בגישה ישירה (Direct Addressing)
המפתח עצמו משמש כאינדקס במערך.

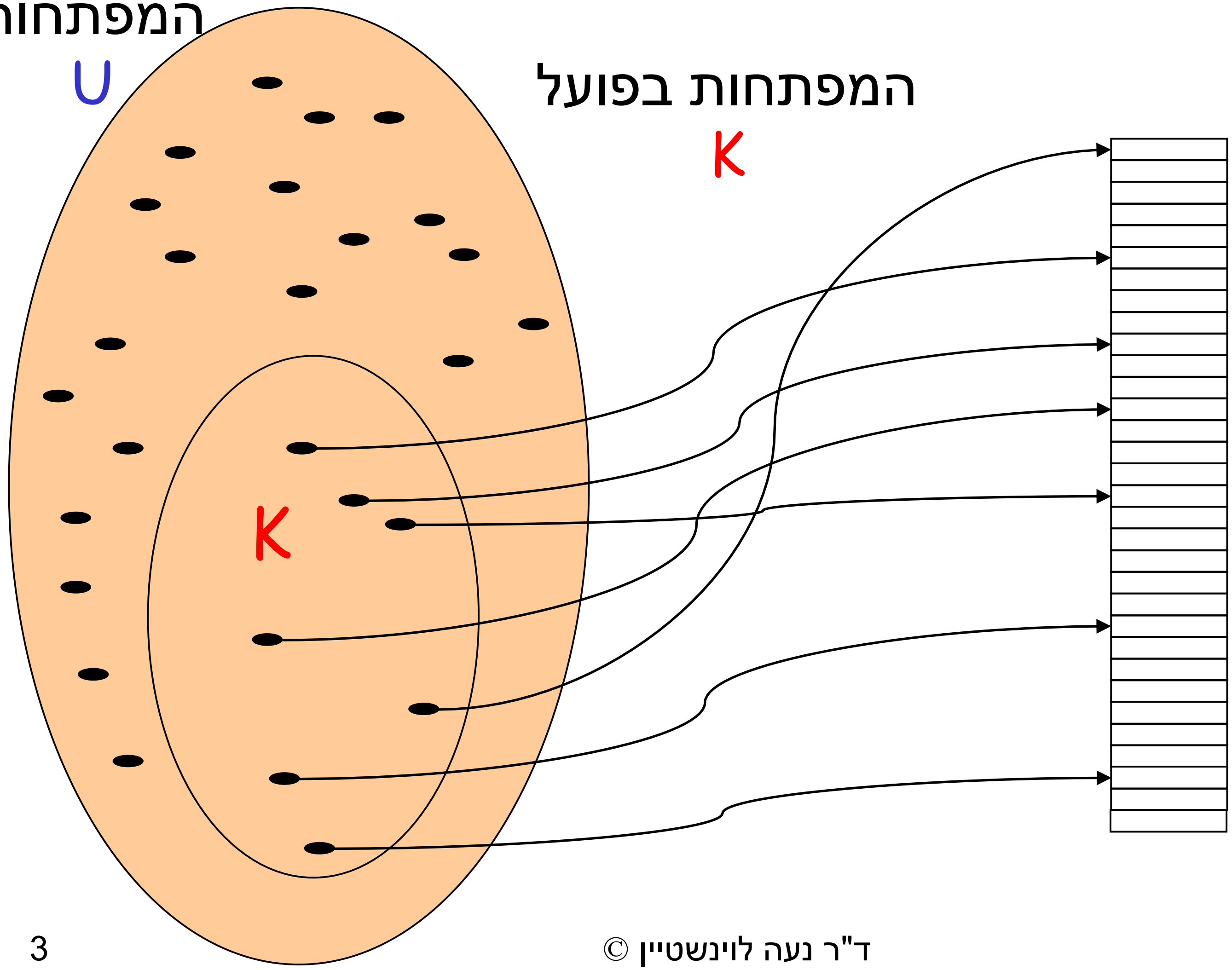


מה עלות החיפוש?

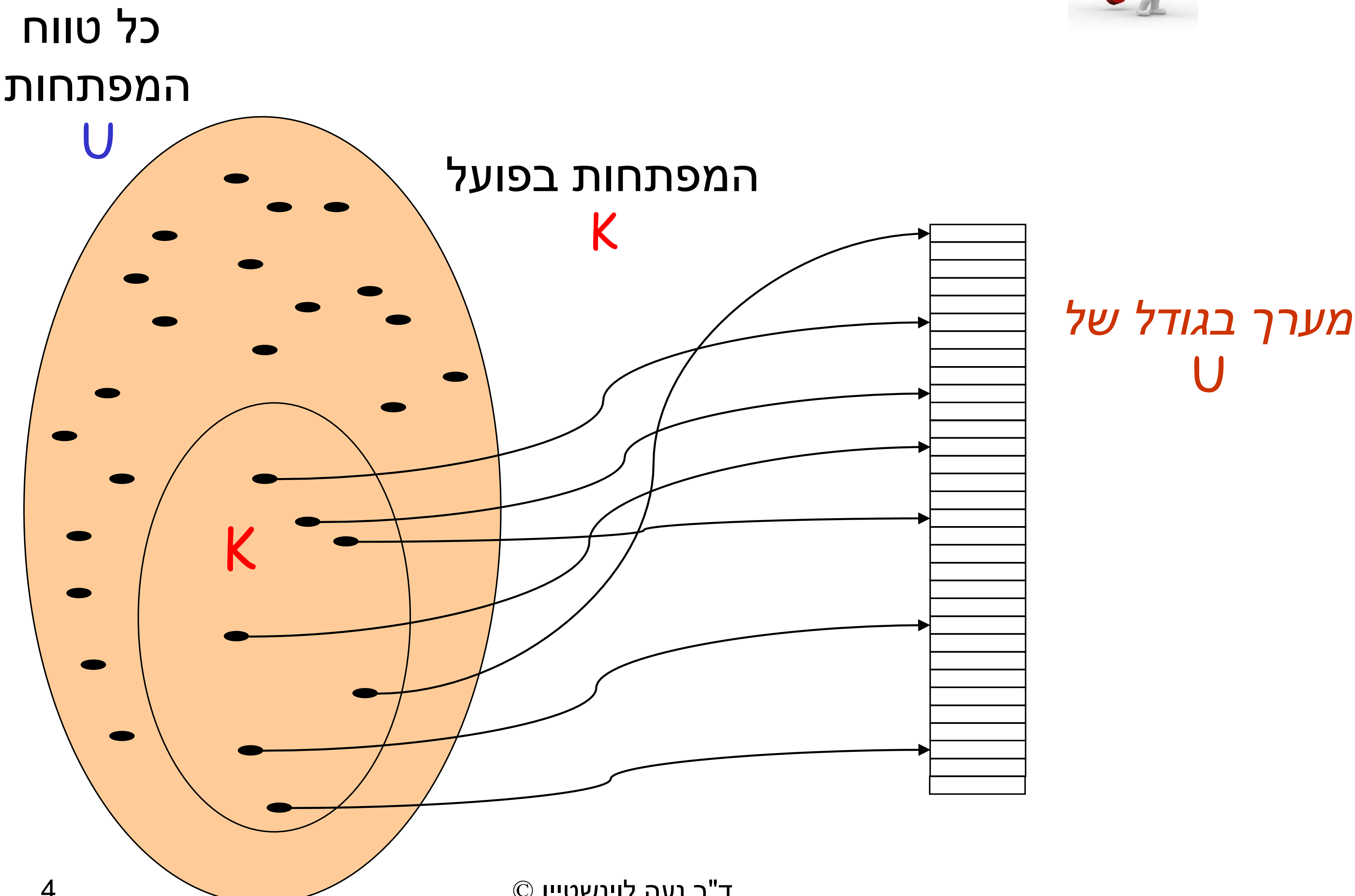


עריכה: ד"ר איילת בוטמן, פרופ' אביבית לוי

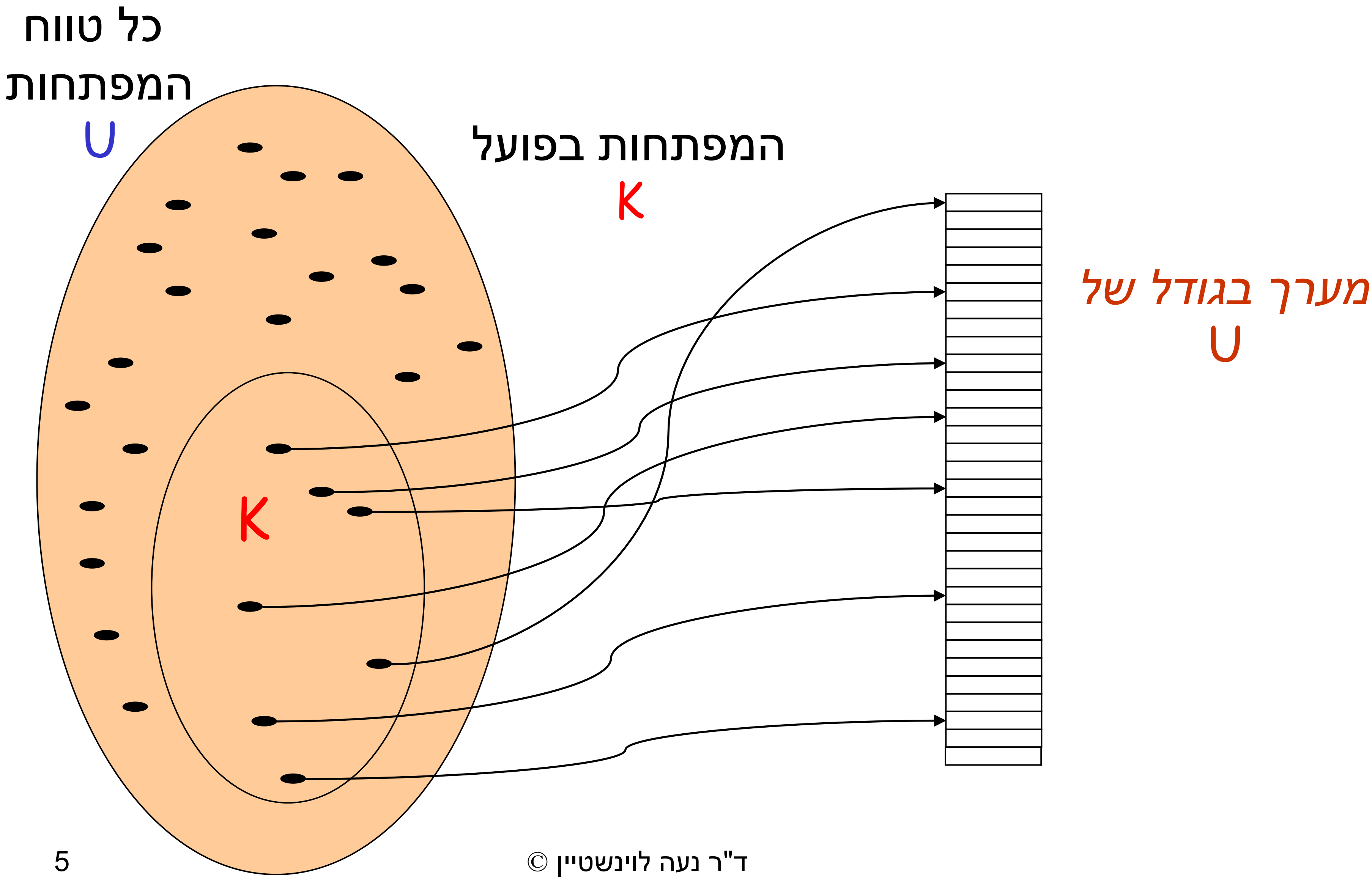
מה עלות החיפוש? O(1)



מה עלות הכנסה?

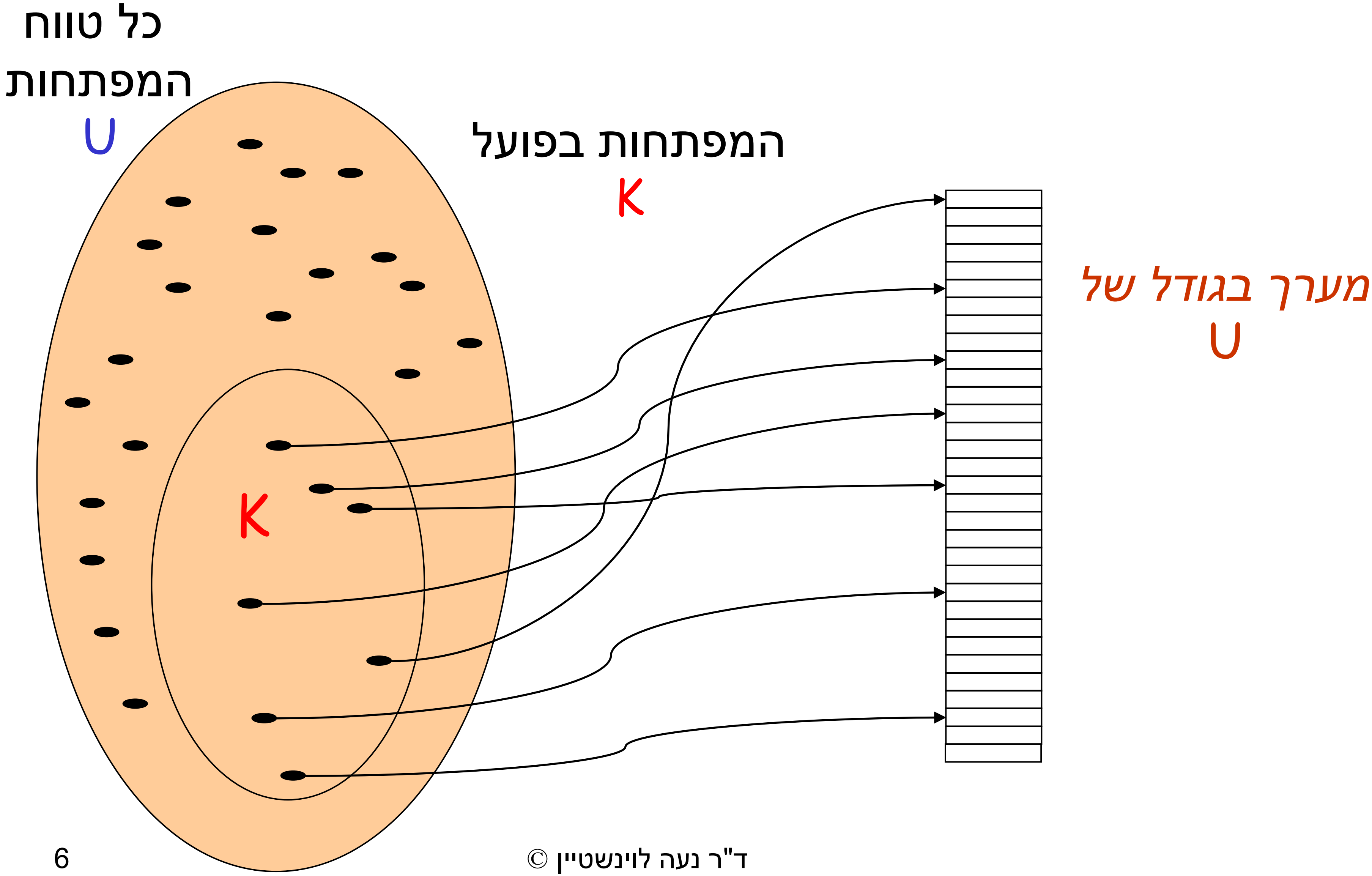


מה עלות הכנסה? O(1)

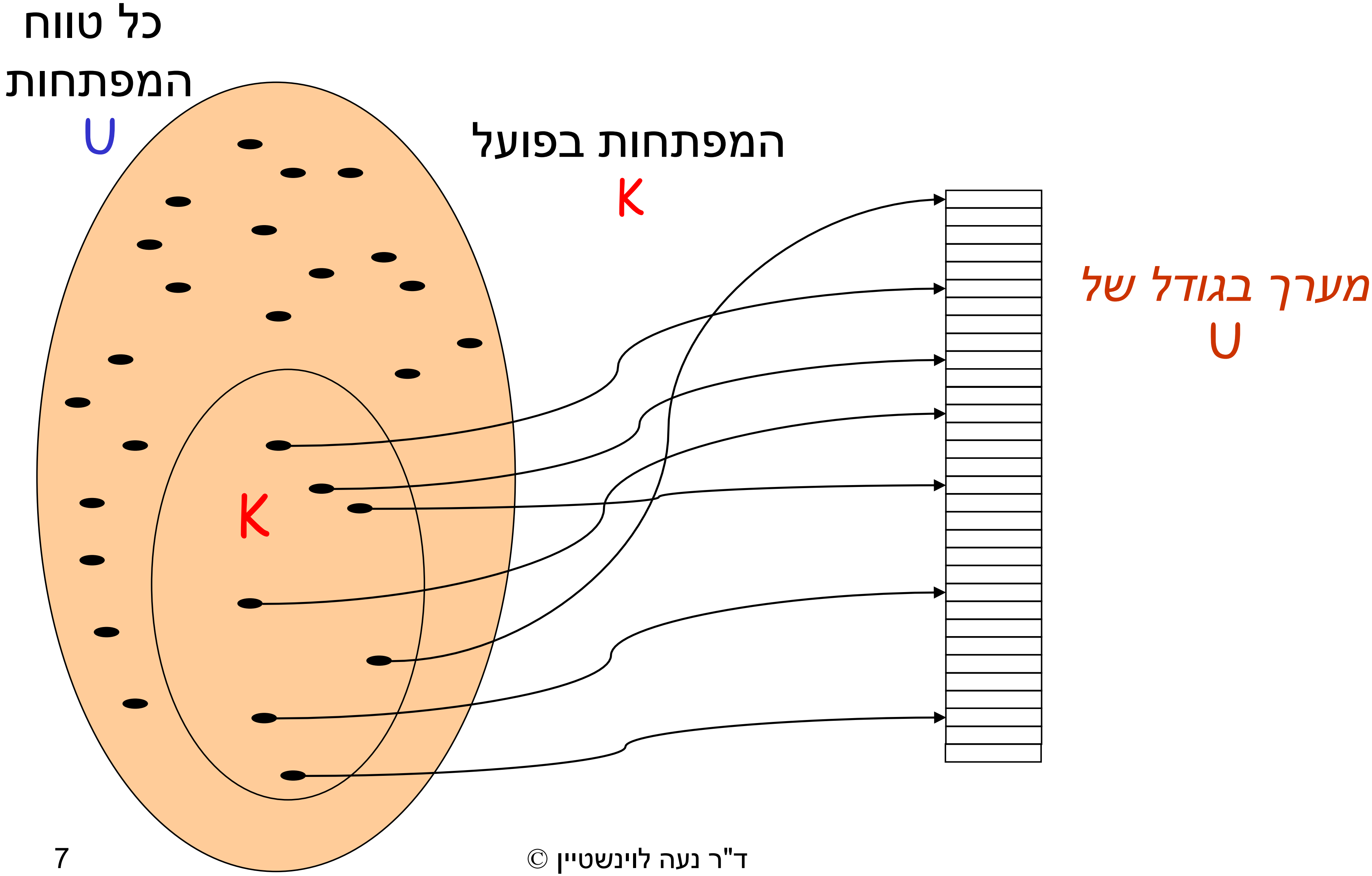


עריכה: ד"ר איילת בוטמן, פרופ' אביבית לוי

מה עלות הוצאה?



מה עלות הוצאה? O(1)



עלות

- שלוש הפעולות (חיפוש, הכנסה והוצאה) מתבצעות בזמן $O(1)$ כאשר משתמשים במערך במיעון ישיר.
- שלוש הפעולות (חיפוש, הכנסה והוצאה) מתבצעות בזמן $O(\log n)$ כאשר משתמשים בעץ חיפוש מאוזן.

מדוע להשתמש בעץ חיפוש מאוזן ?

דוגמא 1

- מספר ת.ז. מורכב מ-9 ספרות עשרוניות.

דוגמא 1

- מספר ת.ז. מורכב מ-9 ספרות עשרוניות.
- כמה מספרי ת.ז. אפשריים?

דוגמא 1

- מספר ת.ז. מורכב מ-9 ספרות עשרוניות.
- כמה מספרי ת.ז. אפשריים? 10^9

דוגמא 1

- מספר ת.ז. מורכב מ-9 ספרות עשרוניות.
- כמה מספרי ת.ז. אפשריים? 10^9
- כמה אזרחים בישראל?

דוגמא 1

- מספר ת.ז. מורכב מ-9 ספרות עשרוניות.
- כמה מספרי ת.ז. אפשריים? 10^9
- כמה אזרחים בישראל? פחות מ-10^7 אנשים.

דוגמא 1

- מספר ת.ז. מורכב מ-9 ספרות עשרוניות.
- כמה מספרי ת.ז. אפשריים? 10^9
- כמה אזרחים בישראל? פחות מ-10^7 אנשים.
- כמה שטח זיכרון ננצל מהמערך?

דוגמא 1

- מספר ת.ז. מורכב מ-9 ספרות עשרוניות.
- כמה מספרי ת.ז. אפשריים? 10^9
- כמה אזרחים בישראל? פחות מ-10^7 אנשים.
- כמה שטח זיכרון ננצל מהמערך? פחות מ-1%

דוגמא 2

- במחרוזת באורך 30 תווים ניתן לתאר:
  - שם פרטי,
  - שם אמצעי,
  - שם משפחה.

- מספר המחרוזות באורך 30 הינו 22^{30} בעוד שבמדינת ישראל יש פחות מ-22^{7} אנשים.

אבחנה

- בפועל אין מערך בגדלים אלו. זמן כל פעולה יהיה $O(1)$ באופן תאורטי בלבד.
- לעומת זאת, בעץ חיפוש מאוזן, 40 $\log_2 n \leq$ כאשר n הוא מספר תושבי מדינת ישראל (או אפילו סין).

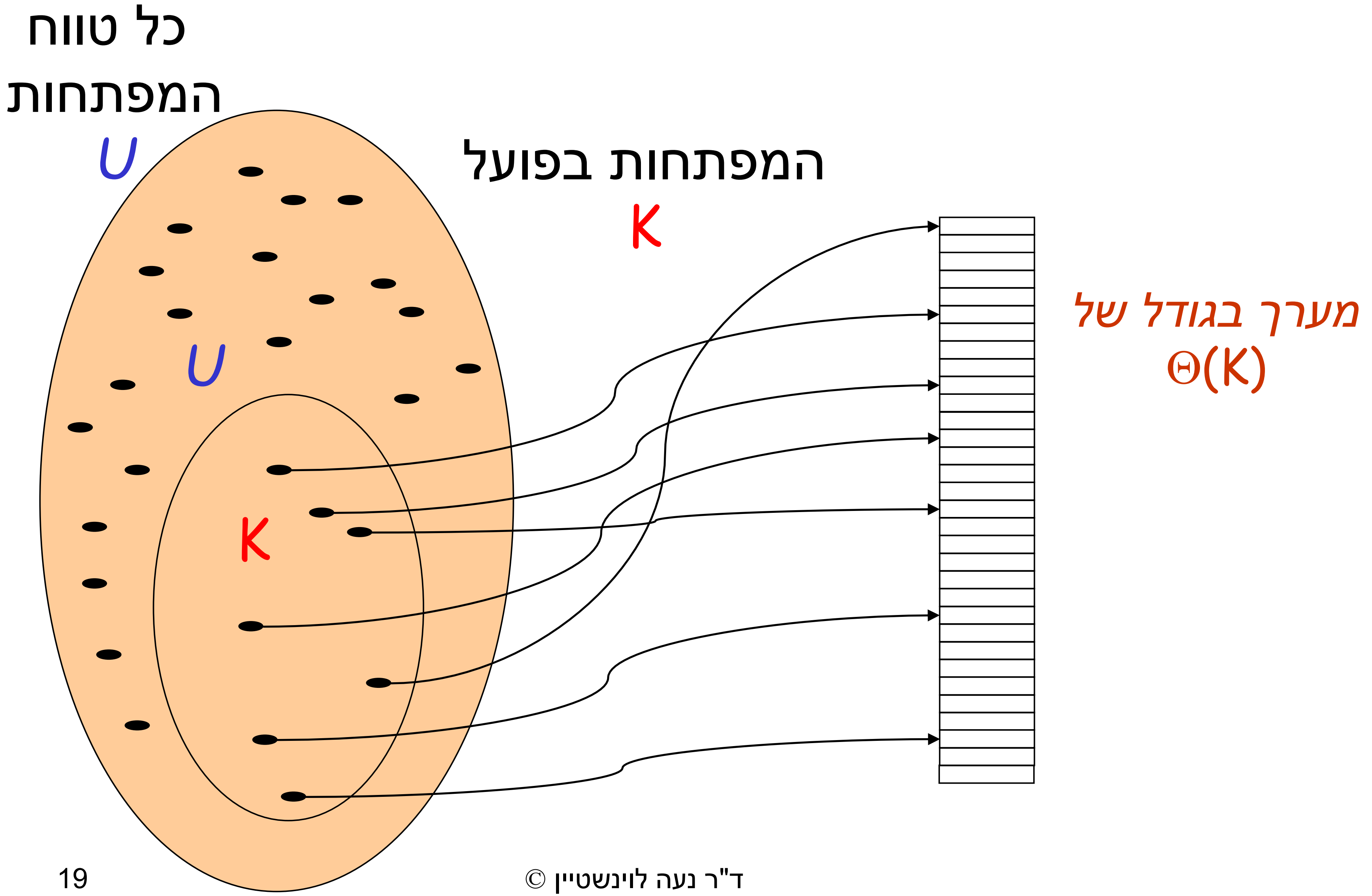
גיבוב Hashing

- כאשר מרחב המפתחות גדול נחשב את האינדקס מתוך המפתח בעזרת פונקצית ערבול/גיבוב.

- **המטרה:**

לממש את הפעולות חיפוש, הכנסה והוצאה בזמן ממוצע של $O(1)$.

טבלת גיבוב



פונקציית ערבול (גיבוב)

$$h: U \rightarrow \{0,1,\ldots,m-1\}$$

- מקבלת מפתח בתחום U ומחשבת אינדקס בטווח המתאים.
- האינדקס של המפתח k הוא $h(k)$.

דוגמא

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

$$h: U \rightarrow \{0,1,\ldots,m-1\}$$

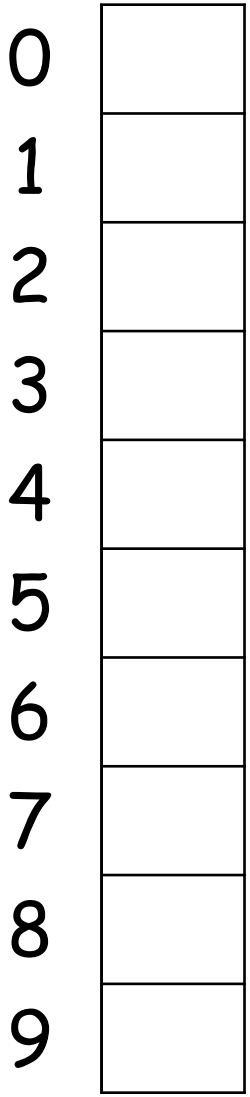
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

דוגמא

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

עבור $U=\{0,1,\ldots,100\}$, $m=10$:

$$h(k) = k \bmod 10$$



דוגמא

$h(k)= k \bmod m$

הכנסת 12:

$h(12) = 12 \bmod 10 = 2$

0	
1	
2	12
3	
4	
5	
6	
7	
8	
9	

דוגמא

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

הכנסת 12:

$$h(12) = 12 \bmod 10 = 2$$

דוגמא

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

הכנסת 17:

$$h(17) = 17 \bmod 10 = 7$$

0	
1	
2	12
3	
4	
5	
6	
7	
8	
9	

0	
1	
2	12
3	
4	
5	
6	
7	17
8	
9	

דוגמא

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

הכנסת 17:

$$h(17) = 17 \bmod 10 = 7$$

0	
1	
2	12
3	
4	
5	
6	
7	17
8	
9	

דוגמא

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

הכנסת 88:

$$h(88) = 88 \bmod 10 = 8$$

0	
1	
2	12
3	
4	
5	
6	
7	17
8	88
9	

דוגמא

$h(k) = k \bmod m$

הכנסת 88:

$h(88) = 88 \bmod 10 = 8$

0	
1	
2	12
3	
4	
5	
6	
7	17
8	88
9	

דוגמא

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

הכנסת 51:

$$h(51) = 51 \bmod 10 = 1$$

דוגמא

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

הכנסת 51:

$$h(51) = 51 \bmod 10 = 1$$

0	
1	51
2	12
3	
4	
5	
6	
7	17
8	88
9	

0	
1	51
2	12
3	
4	
5	
6	
7	17
8	88
9	

דוגמא

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

הכנסת 39:

$$h(39) = 39 \bmod 10 = 9$$

עריכה: ד"ר איילת בוטמן, פרופ' אביבית לוי

0	
1	51
2	12
3	
4	
5	
6	
7	17
8	88
9	39

דוגמא

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

הכנסת 39:

$$h(39) = 39 \bmod 10 = 9$$

0	
1	51
2	12
3	
4	
5	
6	
7	17
8	88
9	39

דוגמא

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

הכנסת 65:

$$h(65) = 65 \bmod 10 = 5$$

דוגמא

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

הכנסת 65:

$$h(65) = 65 \bmod 10 = 5$$

0	
1	51
2	12
3	
4	
5	65
6	
7	17
8	88
9	39

0	
1	51
2	12
3	
4	
5	65
6	
7	17
8	88
9	39

דוגמא

$h(k)= k \bmod m$

הכנסת 81:

$h(81) = 81 \bmod 10 = 1$

0	
1	51
2	12
3	
4	
5	65
6	
7	17
8	88
9	39

דוגמא

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

הכנסת 81:

$$h(81) = 81 \bmod 10 = 1$$

- בשיטת הערבול נוצרות התנגשויות כאשר $x \neq y$ אבל $h(x)=h(y)$.

- לדוגמא, $h(81)=1=h(51)$

0	
1	51
2	12
3	
4	
5	65
6	
7	17
8	88
9	39

דוגמא

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

יש צורך למצוא פתרון לבעיית ההתנגשות

אבל,

ראשית נשתדל לבחור פונקציית גיבוב שהסיכוי להתנגשות יהיה קטן ככל הניתן

0	
1	51
2	12
3	
4	
5	65
6	
7	17
8	88
9	39

דוגמא

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

מה המשמעות ב-mod 10 ?

התחשבות רק בספרת האחדות של המספר!!!

דוגמא

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

0	
1	51
2	12
3	
4	
5	65
6	
7	17
8	88
9	39

מה המשמעות ב-mod 10 ?

התחשבות רק בספרת האחדות של המספר!!!

רצוי שפונקציית הערבול תשתמש בכל המידע שבמספר (ז"א, בכל הספרות)

מסקנה:

לא מומלץ להשתמש ב- $\bmod 10^i$

0	
1	51
2	12
3	
4	
5	65
6	
7	17
8	88
9	39

דוגמא

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

מה המשמעות ב-mod 10 ?

התחשבות רק בספרת האחדות של המספר!!!

רצוי שפונקציית הערבול תשתמש בכל המידע שבמספר (ז"א, בכל הספרות)

מסקנה:

לא מומלץ להשתמש ב- $\bmod 2^i$

למידע
המיוצג בבסיס 2

0	
1	51
2	12
3	
4	
5	65
6	
7	17
8	88
9	39

דוגמא

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

מה המשמעות ב-mod 10 ?

התחשבות רק בספרת האחדות של המספר!!!

רצוי שפונקציית הערבול תשתמש בכל המידע שבמספר (כלומר, בכל הספרות)

המלצה:

בדיקת פונקציית הגיבוב על תת קבוצה של מפתחות "אמיתיים"

עריכה: ד"ר איילת בוטמן, פרופ' אביבית לוי

דוגמאות לפונקציות גיבוב

1. שיטת החילוק: $h(k)= k \bmod m$

m גודל הטבלה. כדאי m ראשוני.

2. שיטת הכפל: $0 \le k < 1$ $h(k)= \lfloor km \rfloor$

3. גיבוב אוניברסלי:

$$h(k)= ((ak+b) \bmod p) \bmod m$$

כאשר p ראשוני גדול מ U ו-a ו-b נבחרים אקראית מהקבוצות $\{1, \cdots, p-1\}$,$\{0, \cdots, p-1\}$ בהתאמה.

התנגשויות

יש לבחור פונקציית גיבוב כך שיווצרו כמה שפחות התנגשויות

הנחת גיבוב אחיד פשוט:

1. ההסתברות שמפתח כלשהו יגובב לתא מסוים **שווה** עבור כל m התאים.
2. חישוב לוקח $O(1)$

נתונה טבלת גיבוב עם m תאים המכילה n איברים.

מקדם העומס של טבלת הגיבוב הוא: $\alpha=n/m$

פתרון ההתנגשויות על-ידי שרשור

מיעון סגור Closed Addressing

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

דוגמא

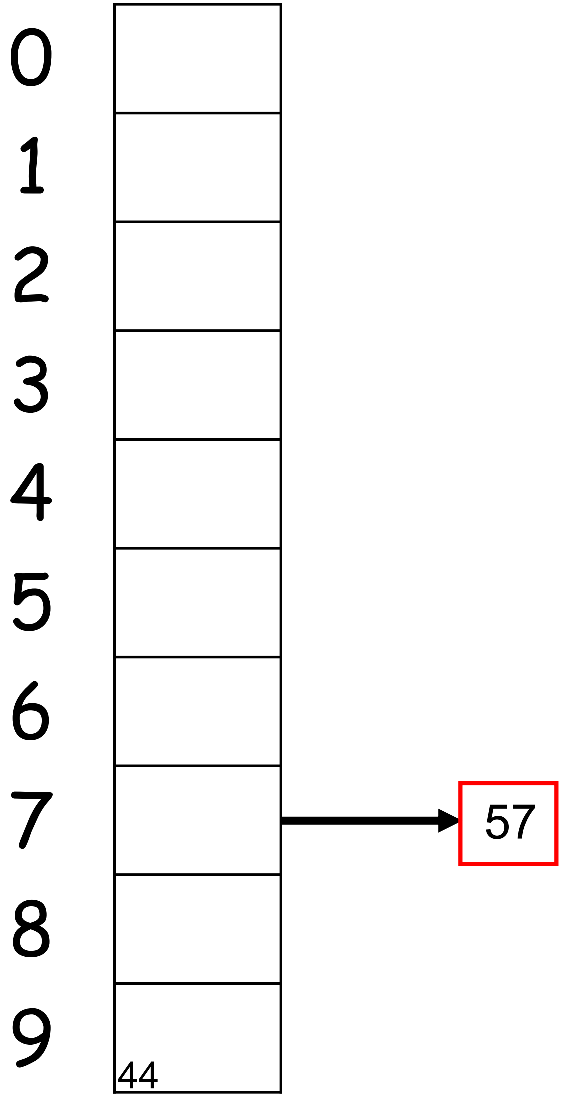
$h(k) = k \bmod 10$,$m=10$, $U=\{0,\ldots 100\}$

- קלט: 57

<u>הערה</u>: $m=10$ נבחר משקולי נוחות ההסבר.

דוגמא

$h(k) = k \bmod 10$,m=10

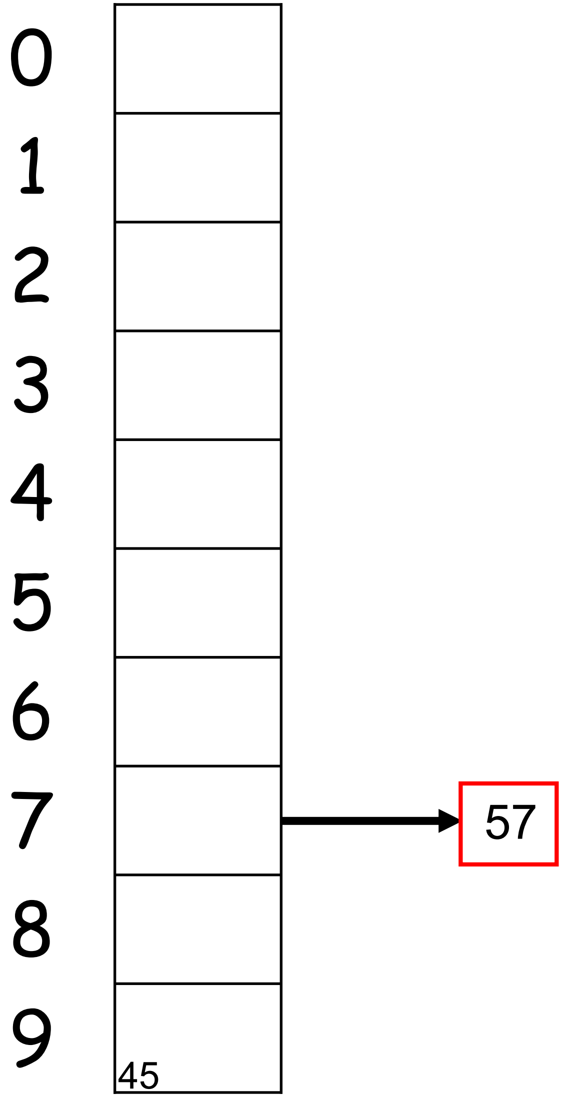


- קלט: 57

דוגמא

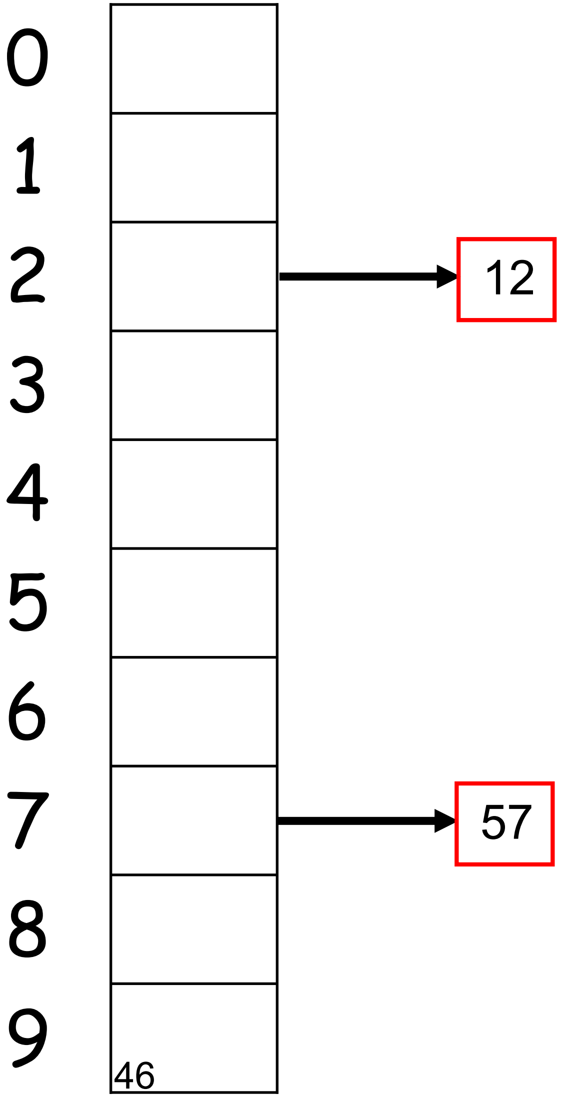
$h(k) = k \bmod 10$,$m=10$

- קלט: 12



דוגמא

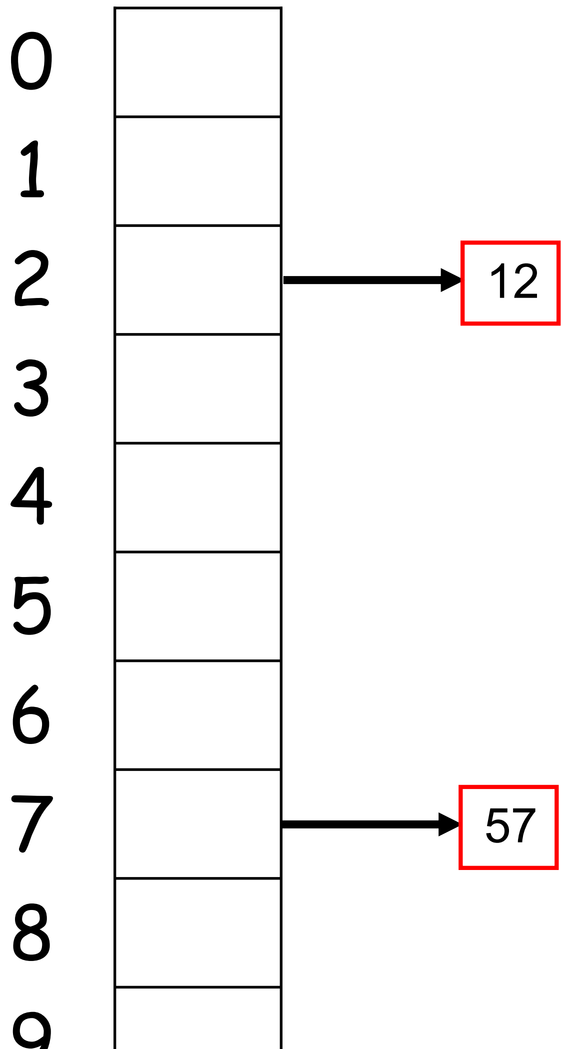
$h(k) = k \bmod 10 \;, m=10$



- קלט: 12

דוגמא

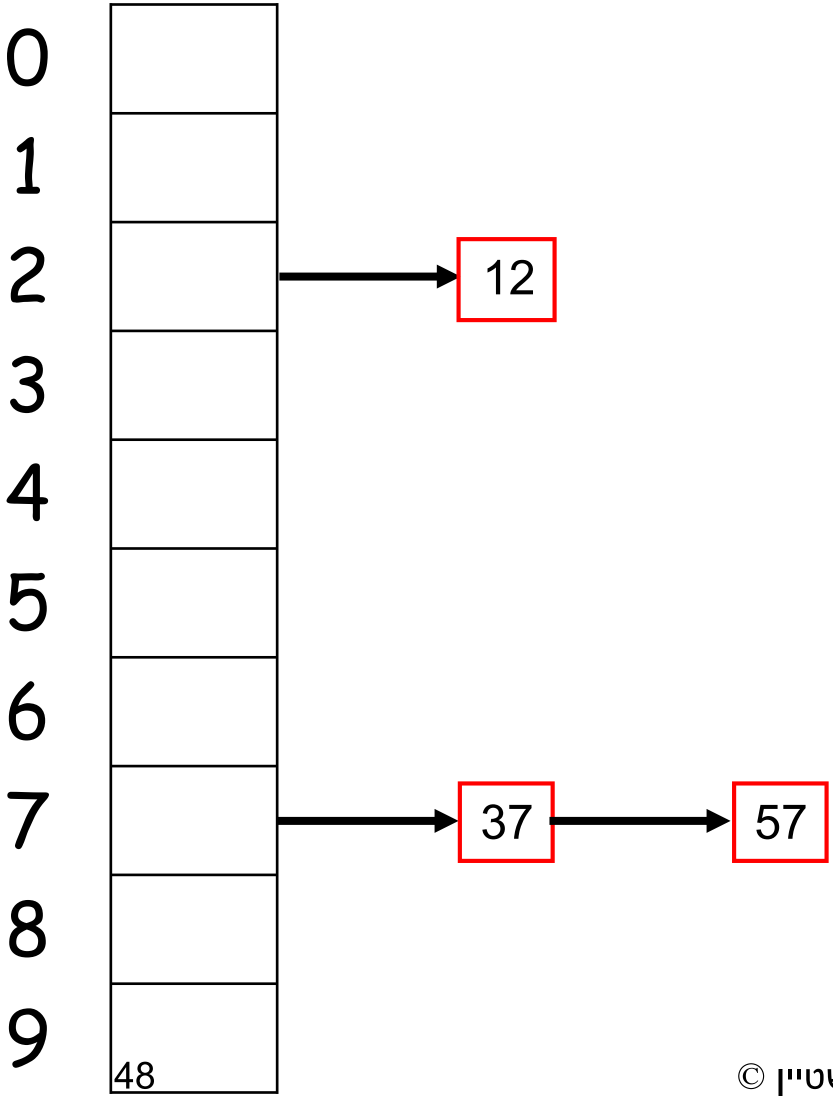
$h(k) = k \bmod 10 \;,m=10$



- קלט: 37

דוגמא

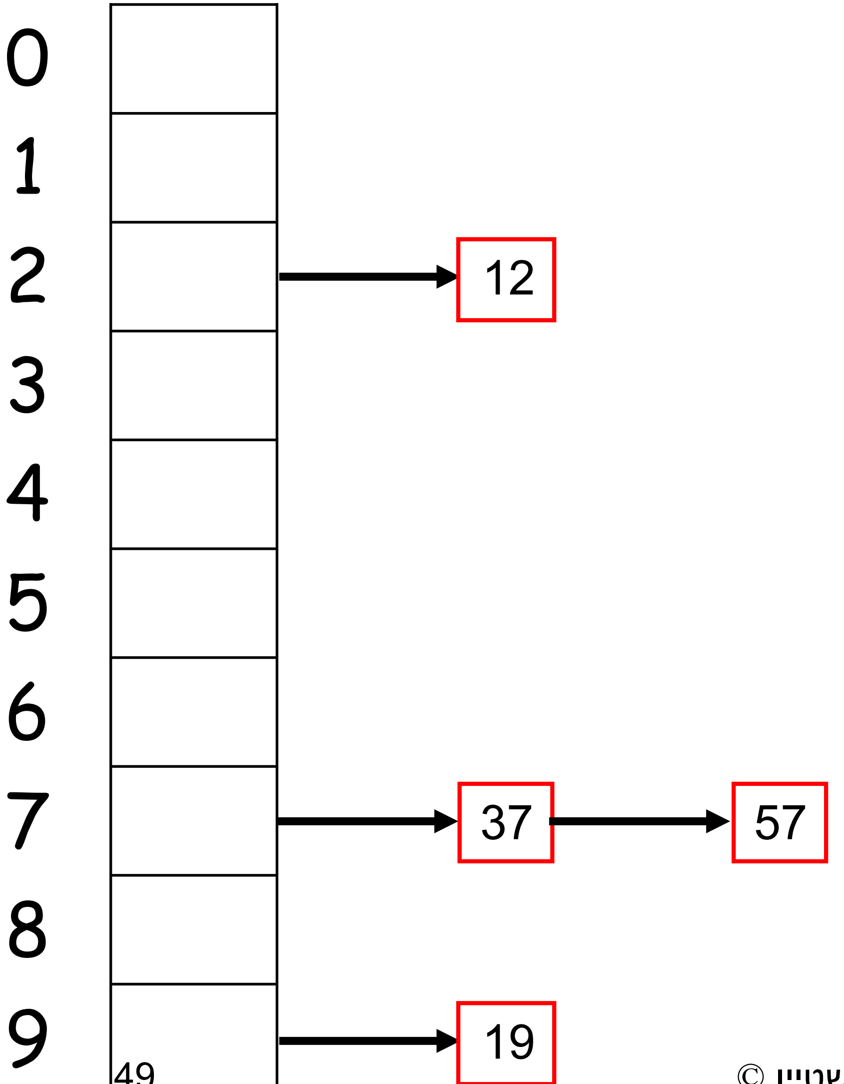
$h(k) = k \bmod 10$,m=10



- קלט: 37

דוגמא

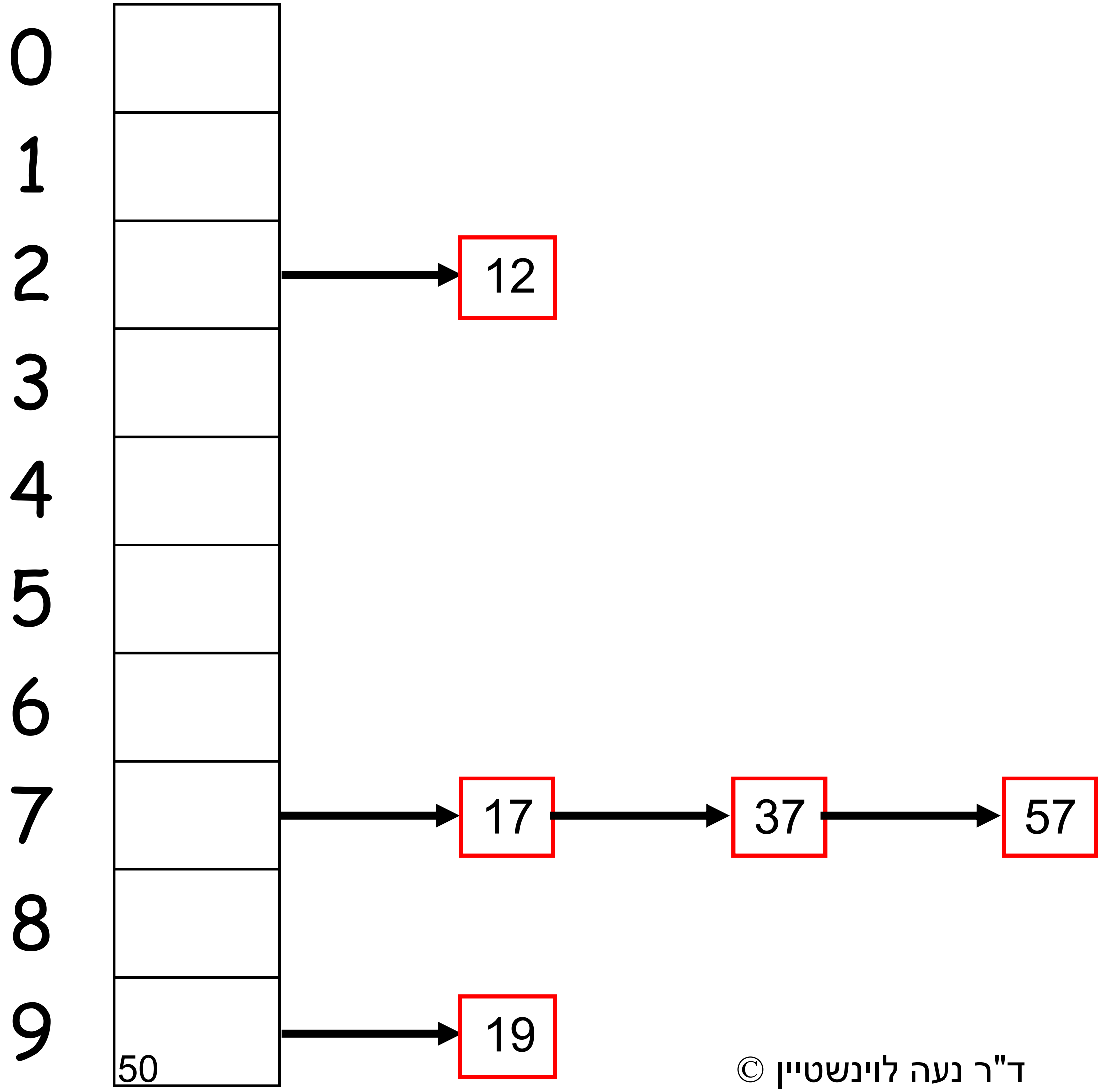
$h(k) = k \bmod 10$,m=10



- קלט: 19

דוגמא

$h(k) = k \bmod 10$,m=10

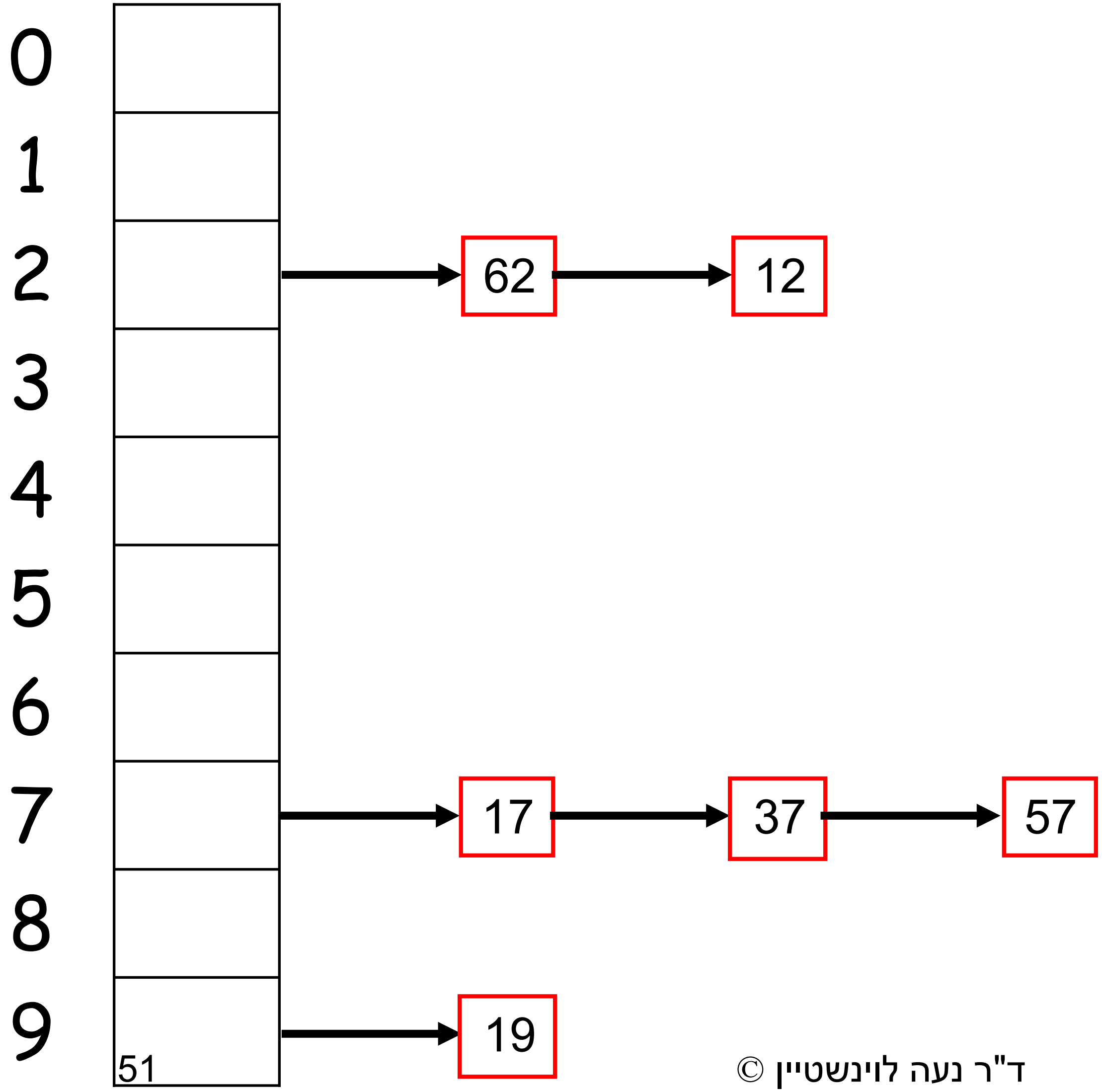


- קלט: 17

דוגמא

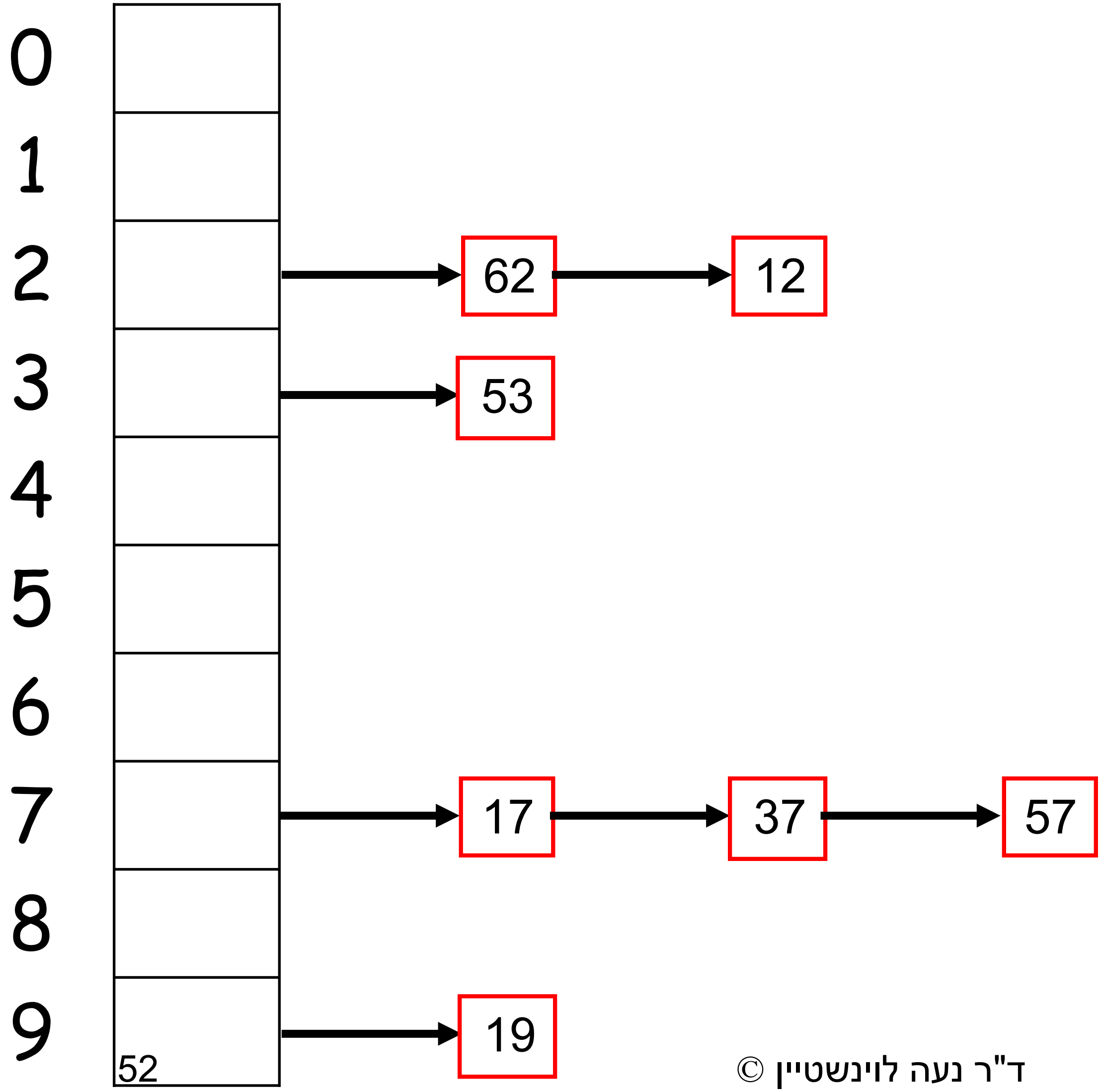
$h(k) = k \bmod 10$,m=10

- קלט: 62



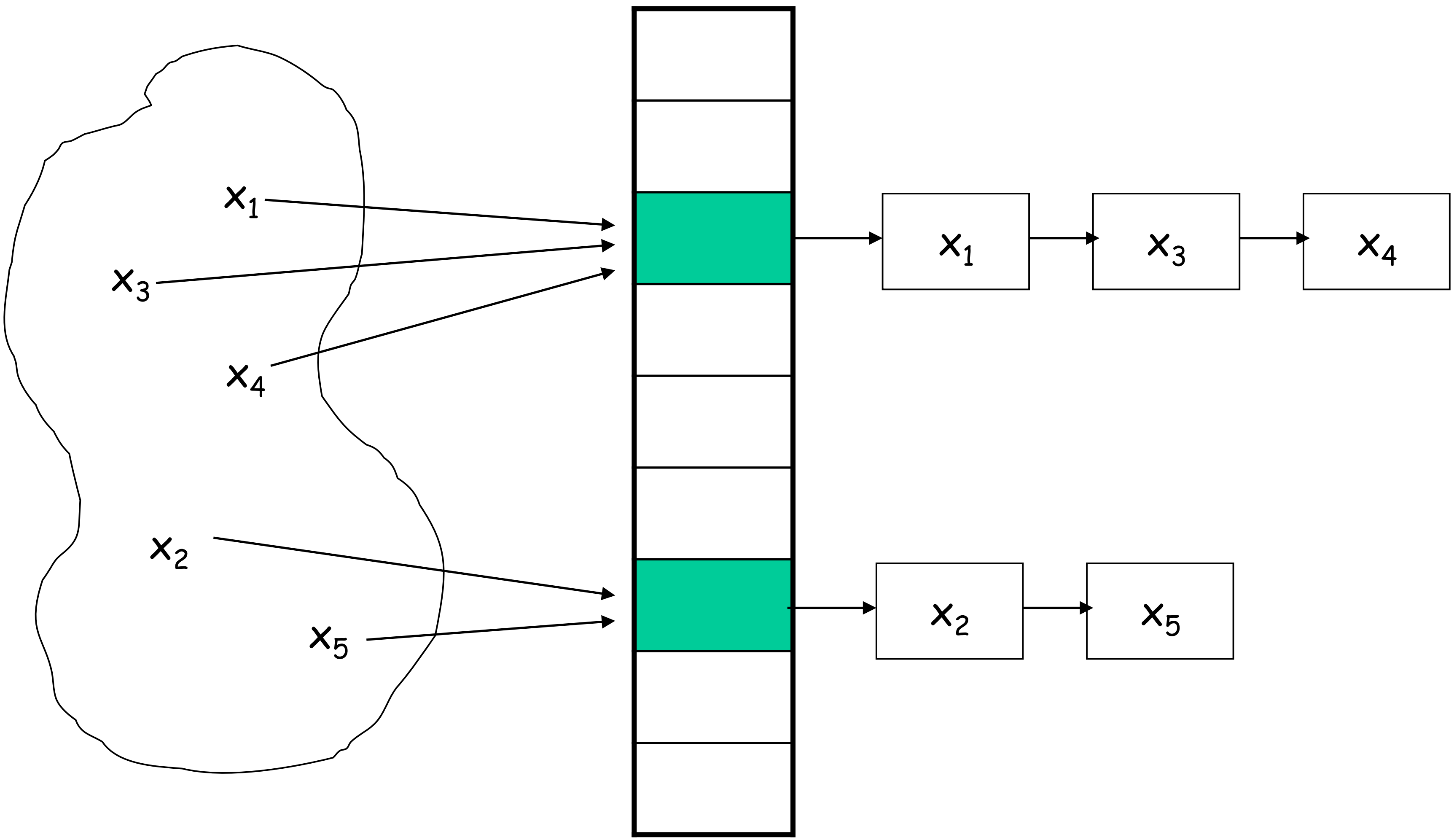
דוגמא

$h(k) = k \bmod 10$,m=10

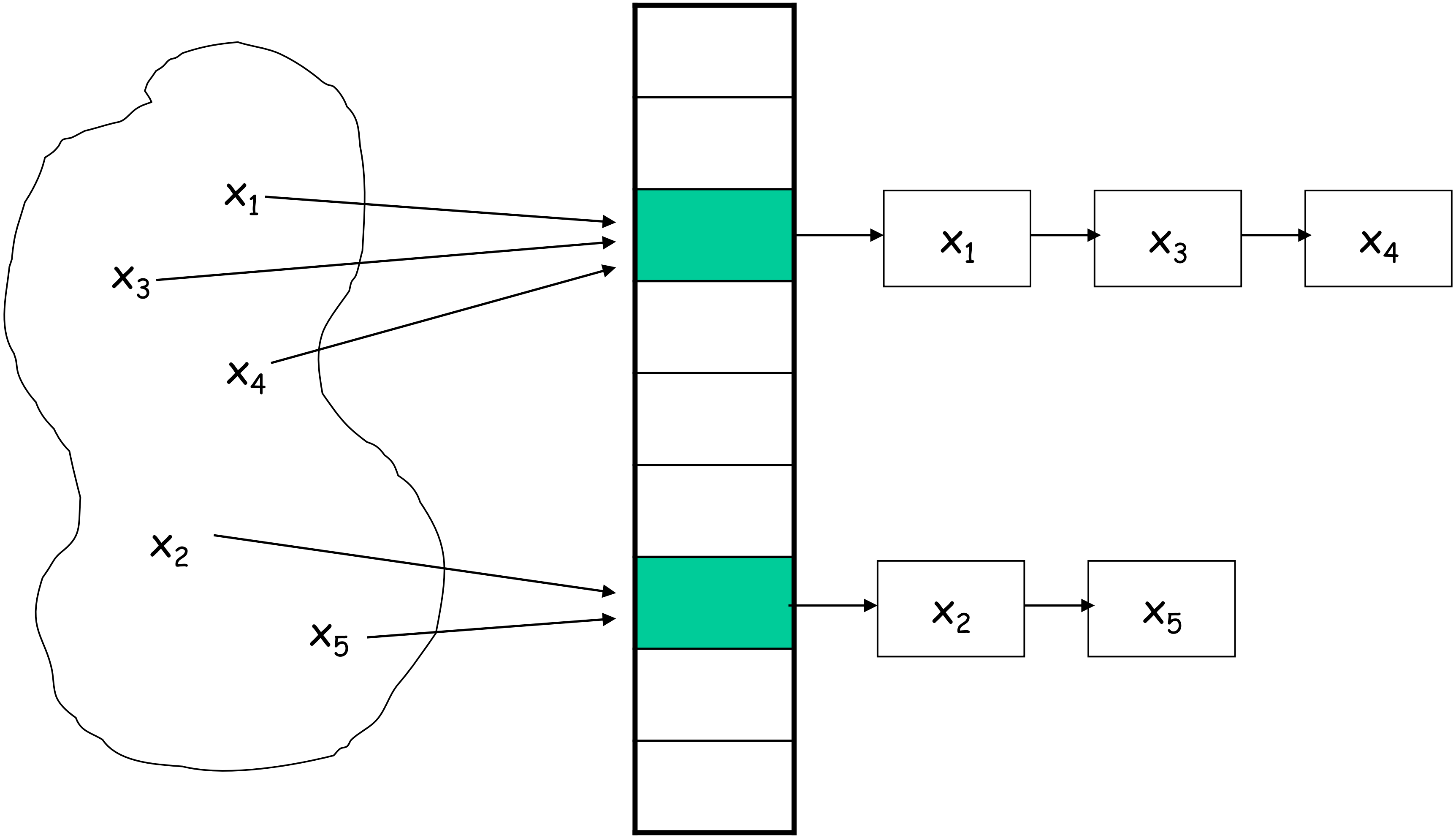


- קלט: 53

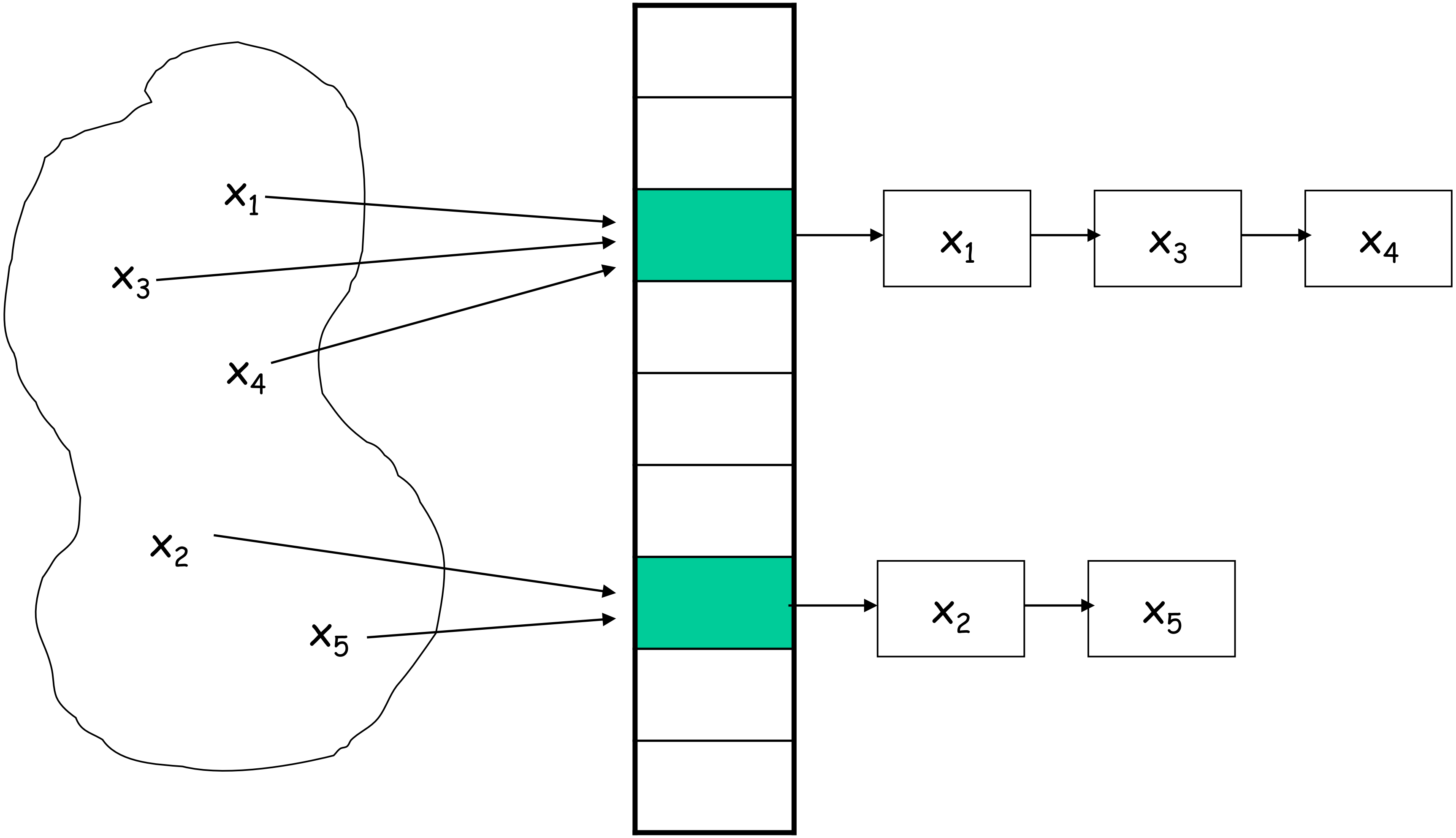
Insert(T,x) הכנס את x מהרשימה.
זמן במקרה הגרוע $O(1)$.



Search(T,k) חפש איבר עם מפתח k ברשימה.
זמן במקרה הגרוע Θ(אורך הרשימה).



Delete(T,x) סלק את x מהרשימה.
זמן במקרה הגרוע (אורך הרשימה)Θ.



Delete(T,x) סלק את x מהרשימה.
זמן במקרה הגרוע θ(אורך הרשימה).

ניתוח זמנים

- במקרה הגרוע כל האיברים נכנסו לאותה רשימה ואז זמן החיפוש/הוצאה הוא $\theta(n)$.
- השימוש בערבול אינו בגלל זמן הביצוע המקסימאלי לפעולה אלא בגלל זמן הביצוע הממוצע לפעולה.
- **המטרה:** לבחור פונקציית ערבול שמפזרת היטב את המפתחות לרשימות השונות.

זמן חיפוש בגיבוב עם שרשור

בהנחה שפונקציית הגיבוב מקיימת את **הנחת גיבוב אחיד פשוט**

במקרה הגרוע: $\theta(n)$

במקרה הממוצע: $\theta(\alpha)$

מסקנה

אם היחס בין מספר התאים בטבלת הגיבוב למספר האיברים בטבלה מקיים: $n=O(m)$, אז

$$\alpha = \frac{n}{m} = \frac{O(m)}{m} = O(1)$$

לכן, חיפוש מתבצע בממוצע בזמן קבוע.

מכיוון שפעולת הכנסה מתבצעת במקרה הגרוע בזמן $O(1)$ והוצאה עולה כמו חיפוש, ניתן לממש את שלושת הפעולות בזמן ממוצע $O(1)$.

דוגמא

עבור 2100 מפתחות מטווח כלשהו U של מספרים שלמים, נאמר עד 10^6,

נוכל להחזיק מערך ובו 700 מקומות

ובממוצע אורך כל שרשרת יהיה 3

וזמני החיפוש יהיו בהתאם.

גיבוב עם שרשור - סיכום

<u>חסרון</u>

דורש מקום נוסף מחוץ לטבלה.

<u>יתרון</u>

הכנסה $O(1)$

מיעון פתוח Open Addressing

טיפול בהתנגשויות על-ידי חיפוש מקום בטבלה עצמה.

דגימה/בדיקה ליניארית

במקרה של התנגשות מאחסנים את האיבר במקום הפנוי הבא בטבלה לפי סדר באופן מעגלי – אחרי התא האחרון עוברים לתא הראשון.

Linear probing דגימה ליניארית

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12

Linear probing דגימה ליניארית

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

ד

0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 18

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 18

$h(18)= 18 \bmod 13 =5$

Linear probing דגימה ליניארית

דוגמא:

ד

פונקציית הגיבוב: $h(k)=k \bmod 13$

					18							
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 18

$h(18)= 18 \bmod 13 =5$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

ד

					18							
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 41

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

					18							
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 41

$h(41)= 41 \bmod 13 =2$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18							

הכנס 41

$h(41)= 41 \bmod 13 =2$

Linear probing דגימה ליניארית

דוגמא:

ד

פונקציית הגיבוב: $h(k)=k \bmod 13$

		41			18							
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס **22**

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18							
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 22

$h(22)= 22 \bmod 13 =9$

Linear probing דגימה ליניארית

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18				22			
0	1	2	3	4	5	6	7	8	9	10	11	12

22 הכנס

$h(22)= 22 \bmod 13 =9$

דגימה ליניארית   Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18				22			

הכנס 31

$h(31)= 31 \bmod 13 =5$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			**18**				22			

הכנס 31

$h(31)= 31 \bmod 13 =5$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31			22			
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס **31**

$h(31)= 31 \bmod 13 =5$

Linear probing דגימה ליניארית

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31			22			
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 46

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31			22			

הכנס 46

$h(46)= 46 \bmod 13 =7$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46		22			

הכנס 46

$h(46)= 46 \bmod 13 =7$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31	46		22			
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס **58**

Linear probing דגימה ליניארית

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46		22			

הכנס 58

$h(58)= 58 \bmod 13 =6$

Linear probing   דגימה ליניארית

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46		22			

הכנס 58

$h(58)= 58 \bmod 13 =6$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: h(k)=k mod 13

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46		22			

הכנס **58**

h(58)= 58 mod 13 =6

דגימה ליניארית Linear probing

דוגמא:

T

פונקציית הגיבוב: $h(k)=k \bmod 13$

		41			18	31	46	58	22			
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 58

$h(58)= 58 \bmod 13 =6$

Linear probing דגימה ליניארית

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31	46	58	22			
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 58

$h(58)= 58 \bmod 13 =6$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46	58	22			

הכנס 20

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46	58	22			

הכנס 20

$h(20)= 20 \bmod 13 =7$

דגימה ליניארית   Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31	46	58	22			
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 20

$h(20)= 20 \bmod 13 =7$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31	46	58	22	20		
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 20

$h(20)= 20 \bmod 13 =7$

דגימה ליניארית   Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31	46	58	22	20		
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 44

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31	46	58	22	20		
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס **44**

$h(44)= 44 \bmod 13 =5$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31	46	58	22	20		
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס **44**

$h(44)= 44 \bmod 13 =5$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46	58	22	20	44	

הכנס **44**

$h(44)= 44 \bmod 13 =5$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31	46	58	22	20	44	
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 25

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46	58	22	20	44	

הכנס 25

$h(25)= 25 \bmod 13 =12$

דגימה ליניארית   Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

		41			18	31	46	58	22	20	44	25
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 25

$h(25)= 25 \bmod 13 =12$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46	58	22	20	44	25

הכנס **21**

דגימה ליניארית   Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
		41			18	31	46	58	22	20	44	25

הכנס 21

$h(21)= 21 \bmod 13 =8$

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: h(k)=k mod 13

T

		41			18	31	46	58	22	20	44	25
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 21

h(21)= 21 mod 13 =8

דגימה ליניארית Linear probing

דוגמא:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

21		41			18	31	46	58	22	20	44	25
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס 21

$h(21)= 21 \bmod 13 =8$

דגימה ליניארית Linear probing

דוגמא:

T

פונקציית הגיבוב: $h(k)=k \bmod 13$

21		41			18	31	46	58	22	20	44	25
0	1	2	3	4	5	6	7	8	9	10	11	12

הכנס **21**

$h(21)= 21 \bmod 13 = 8$

מה יקרה אם נרצה
למחוק את 58?

דוגמאות לשימוש במילון ללא הוצאות

- טבלה של שמות משתנים בהרצת תוכנית (Symbol Table).
- מספרי תעודות זהות אינם ממוחזרים.

מיעון פתוח Open Addressing

חיפוש

נניח ש- T טבלת גיבוב עם דגימה ליניארית

חיפוש איבר בעל מפתח k

- הפעל את פונקצית הגיבוב h(k)
- עבור על התאים מהתא עם אינדקס h(k) עד לאחד מהמקרים הבאים:

1. נמצא איבר עם מפתח k - החזר אותו
2. הגעת לתא ריק - האיבר לא נמצא
3. עברת על m תאים - האיבר לא נמצא

חיפוש

```
find(k)

i ← h(k)
p ← 0
repeat
      c ← A[i]
       if  c  =∅ then
           return(NO_SUCH_KEY)
      else  if key(c)=k  then
                       return(element(c))
                else
                       i ← (i+1) mod m
                       p ← p+1
until  p=m
return(NO_SUCH_KEY)
```

הוצאה

נניח שטבלת גיבוב T עם דגימה ליניארית

<u>הוצאת איבר בעל מפתח k</u>

- חפש איבר עם מפתח k
- אם נמצא – הצב בתא זה ערך "ריק" והחזר את האיבר אחרת, החזר "לא קיים"

הוצאה

נניח שטבלת גיבוב T עם דגימה ליניארית

<u>הוצאת איבר בעל מפתח k</u> (בשיטת המצבה)

- חפש איבר עם מפתח k
- אם נמצא – הצב בתא זה ערך "נמחק" והחזר את האיבר אחרת, החזר "לא קיים"

Linear probing דגימה ליניארית

דוגמא להוצאה בשיטת המצבה:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
21		41			18	31	46	58	22	20	44	25

הוצא 58

$h(58)= 58 \bmod 13 =6$

דגימה ליניארית Linear probing

דוגמא להוצאה בשיטת המצבה:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

0	1	2	3	4	5	6	7	8	9	10	11	12
21		41			18	31	46	58	22	20	44	25

הוצא 58

$h(58)= 58 \bmod 13 =6$

דגימה ליניארית Linear probing

דוגמא להוצאה בשיטת המצבה:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

21		41			18	31	46	58	22	20	44	25
0	1	2	3	4	5	6	7	8	9	10	11	12

הוצא 58

$h(58)= 58 \bmod 13 =6$

דגימה ליניארית Linear probing

דוגמא להוצאה בשיטת המצבה:

פונקציית הגיבוב: $h(k)=k \bmod 13$

T

21		41			18	31	46	58	22	20	44	25
0	1	2	3	4	5	6	7	8	9	10	11	12

הוצא 58

$h(58)= 58 \bmod 13 =6$

דגימה ליניארית Linear probing

דוגמא להוצאה בשיטת המצבה:

פונקציית הגיבוב: $h(k)=k \bmod 13$

ד

21		41			18	31	46	delete	22	20	44	25
0	1	2	3	4	5	6	7	8	9	10	11	12

הוצא 58

$h(58)= 58 \bmod 13 =6$

יתרונות וחסרונות לשיטת המצבה

- יתרון: פשוט לביצוע.
- חסרון: כאשר השימוש דורש הוצאות, אורך החיפוש תלוי גם באיברים שכבר הוצאו ולא רק באיברים שכרגע במבנה.

הכנסה בשיטת המצבה

בהינתן טבלת גיבוב T עם דגימה ליניארית,
נניח שהטבלה לא מלאה.

<u>הכנסת איבר בעל מפתח k</u>

- הפעל את פונקציית הגיבוב h(k)
- עבור על התאים מהתא עם אינדקס h(k) (באופן מעגלי) עד שמגיעים לתא ריק או לתא המכיל ערך "נמחק"
  - הכנס את האיבר

בדיקה לינארית: סיכום

- יתרון - בדיקה ליניארית קלה למימוש.
- חיסרון - יתכנו רצפים ארוכים של תאים תפוסים, שמאריכים את זמן החיפוש הממוצע. זוהי תופעת הצטברות ראשונית.

גיבוב כפול Double Hashing

מחפשים מקום פנוי בטבלה בקפיצות משתנות (ולא לפי סדר).

איך?

בוחרים פונקצית גיבוב נוספת $d(k)$ ומחפשים מקום פנוי בטבלה בקפיצות של $d(k)$,

כלומר, הקפיצות יהיו $jd(k)$ עבור $j=0,1,\ldots,m-1$ באופן מעגלי.

$$g(k,j)=(h(k)+ jd(k)) \bmod m$$

0	
1	79
2	
3	
4	69
5	98
6	
7	72
8	
9	
10	
11	50
12	

דוגמא:

$h(k) = k \bmod 13$

$d(k) = 1 + (k \bmod 11)$

הכנס: 14.

0	
1	79
2	
3	
4	69
5	98
6	
7	72
8	
9	
10	
11	50
12	

דוגמא:

$h(k)= k \bmod 13$

$d(k)=1+(k \bmod 11)$

$h(14)= 14 \bmod 13 = 1$

0	
1	79
2	
3	
4	69
5	98
6	
7	72
8	
9	
10	
11	50
12	

דוגמא:

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

$$d(k) = 1 + (k \bmod 11)$$

$$h(14) = 14 \bmod 13 = 1$$

0	
1	79
2	
3	
4	69
5	98
6	
7	72
8	
9	14
10	
11	50
12	

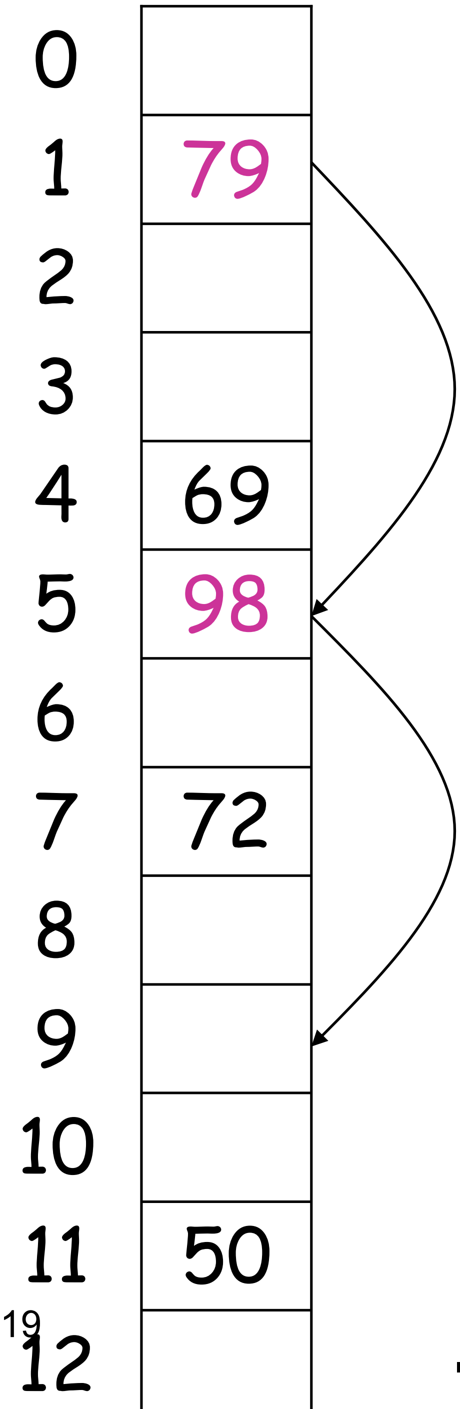
דוגמא:

$h(k) = k \bmod 13$

$d(k) = 1 + (k \bmod 11)$

$h(14) = 14 \bmod 13 = 1$

$d(14) = 1 + (14 \bmod 11) = 4$



דוגמא:

$h(k)= k \bmod 13$

$d(k)=1+(k \bmod 11)$

$h(14)= 14 \bmod 13 = 1$

$d(14)= 1 + (14 \bmod 11) = 4$

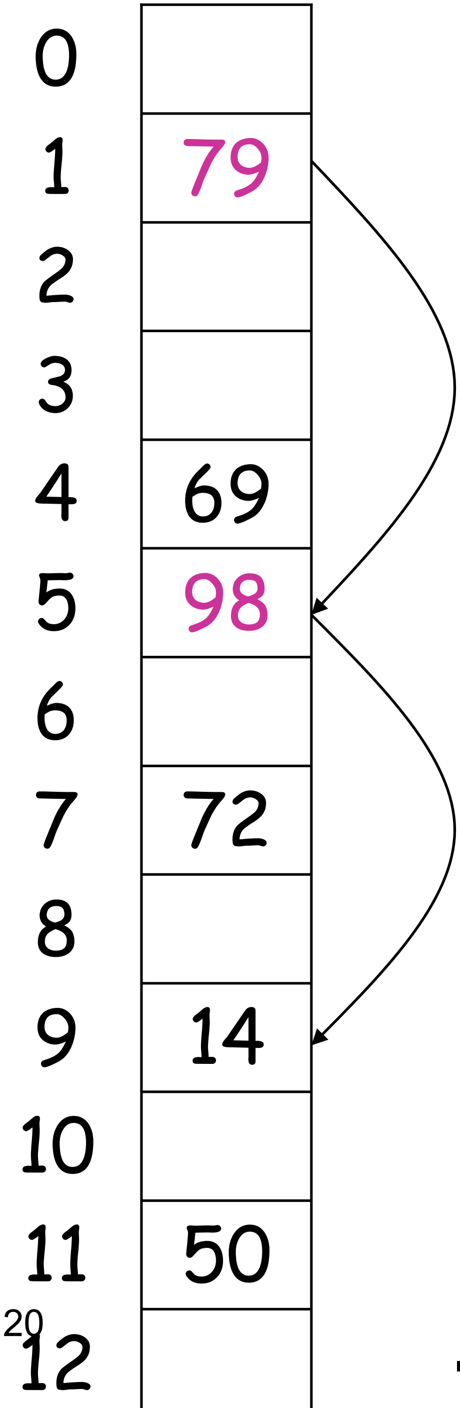
דוגמא:

$h(k)= k \bmod 13$

$d(k)=1+(k \bmod 11)$

$h(14)= 14 \bmod 13 = 1$

$d(14)= 1 + (14 \bmod 11) = 4$



הערה

- כדי שהחיפוש אחר תא פנוי יסרוק את טבלת הגיבוב כולה, הערך $d(k)$ חייב להיות זר לגודל m של טבלת הגיבוב.
- פתרון נוח: בחירת m ראשוני, ובניית d כך שתפיק תמיד מספר שלם חיובי קטן מ-m.

עריכה: ד"ר איילת בוטמן, פרופ' אביבית לוי

דוגמא

$h(k)= k \bmod m$
$d(k)=1+(k \bmod m')$ $m'=m-1$

$K=123456,\ m=701,\ m'=700$

$\rightarrow 80,\ 80 + 257$

משפט

- בהינתן טבלת גיבוב עם מיעון פתוח, שמקדם העומס שלה $\alpha=n/m<1$, ובהנחת הגיבוב האחיד, תוחלת מספר הבדיקות הנערכות בעת חיפוש כושל היא לכל היותר $1/(1-\alpha)$
- מסקנה: אם α קבוע, חיפוש כושל מתבצע בזמן $O(1)$.

מסקנה

- הכנסת איבר לטבלת גיבוב עם מיעון פתוח, בעלת מקדם עומס α, דורשת בממוצע $1/(1-\alpha)$ בדיקות לכל היותר, בהנחת גיבוב אחיד פשוט.

משפט

- בהינתן טבלת גיבוב עם מיעון פתוח, בעלת מקדם עומס $\alpha<1$, תוחלת מספר הבדיקות הנערכות בעת חיפוש מוצלח היא לכל היותר:

$$1/\alpha \ln(1/(1-\alpha))$$

בהנחת גיבוב אחיד פשוט, ובהנחה שההסתברות לחיפוש מפתח מסוים שווה עבור כל המפתחות בטבלה.

קצת מספרים להשוואה...

טבלת תוחלת מספר הבדיקות בחיפוש בבדיקה לינארית

גורם העומס	1/2	2/3	3/4	9/10
חיפוש מוצלח (hit)	1.5	2.0	3.0	5.5
חיפוש לא-מוצלח (miss)	2.5	5.0	8.5	55.5

טבלת תוחלת מספר הבדיקות בחיפוש בגיבוב כפול

גורם העומס	1/2	2/3	3/4	9/10
חיפוש מוצלח (hit)	1.4	1.6	1.8	2.6
חיפוש לא-מוצלח (miss)	1.5	2.0	3.0	5.5

טבלאות גיבוב דינמיות
Dynamic Hash Tables

- העלות בממוצע לפעולת הכנסה נשארת $O(1)$, למרות שיש פעולה יקרה של הרחבת הטבלה כאשר גורם העומס עולה מעל לגודל שנקבע.
- **ניתוח לשיעורין (Amortized Analysis)** מראה חסם במקרה הגרוע על עלות ההכנסה במקרה הממוצע:

כאשר מתבצעת פעולת הרחבה ידוע כי α גדול מ $\frac{1}{2}$, ובהכרח קדמו לפעולת ההרחבה היקרה (שעולה N), N פעולות הכנסה ללא הרחבה (שעולות 1) שבהן גורם העומס לא עבר את הסף המותר.

מכאן שהעלות הממוצעת לפעולה (כולל פעולת הרחבה) היא:

$$\frac{\text{עלות}}{\text{פעולות}} = \frac{N\cdot 1+1\cdot N}{N+1} = \frac{2\cdot N}{N+1} < \frac{2\cdot N}{N} = 2$$