

ארגון המחשב ומערכות הפעלה

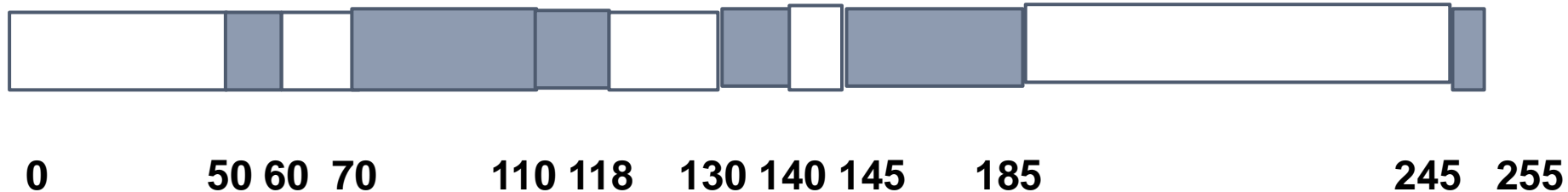
אביב תשפ"ד

תרגול 12 – הקצאות זיכרון וזיכרון וירטואלי

מדיניות הקצאת זיכרון (Malloc)

להלן שרטוט שמתאר את מצב ה Heap במערכת התומכת בהקצאה רציפה של זיכרון (contiguous allocation).

- האזורים הכהים מייצגים שטחים שהוקצו
- הכתובות נתונות ביחידות של 1KByte



מדיניות הקצאת זיכרון (Malloc)

כעת מגיעה בקשה להקצאה של 3KByte, ולאחריה בקשה ל 49KByte.
אילו אזורי זיכרון יוקצו (אם יוקצו) עבור כל אחת מהבקשות אם משתמשים בכל אחת מהשיטות הבאות:

1. First-Fit

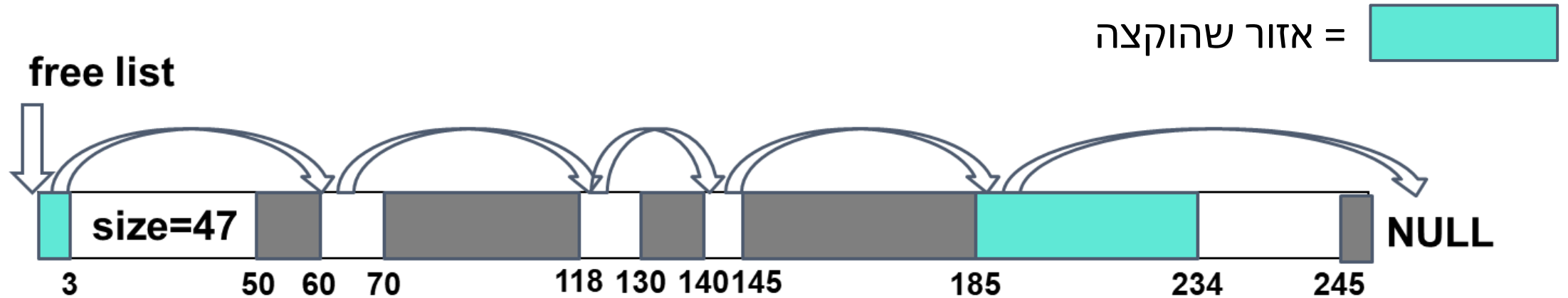
2. Next-Fit

3. Best-Fit

4. Worst-Fit

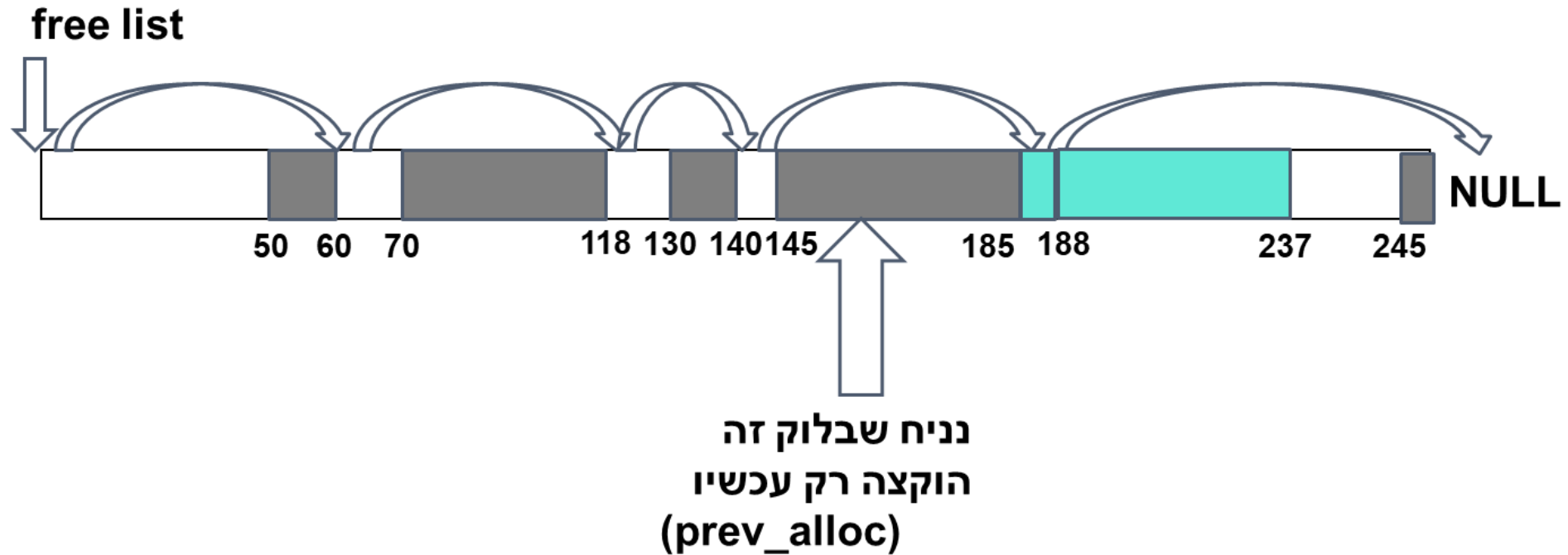
ציינו כתובת התחלה וסיום או כישלון.

First-Fit



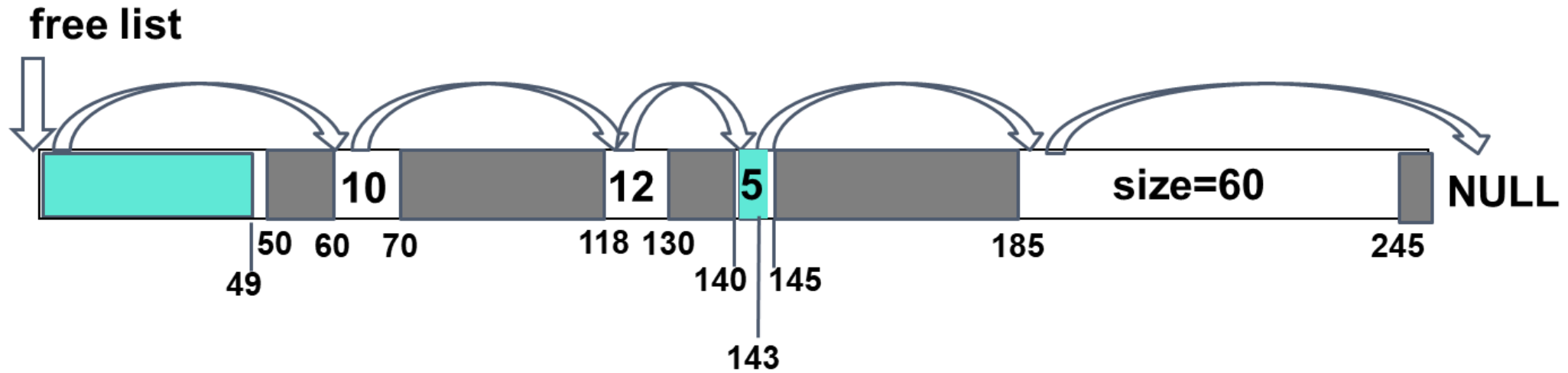
- הקצאה במקום הראשון שיש בו מקום.
- מדידות מראות שזה הכי מהיר והכי חסכוני.

Next-fit



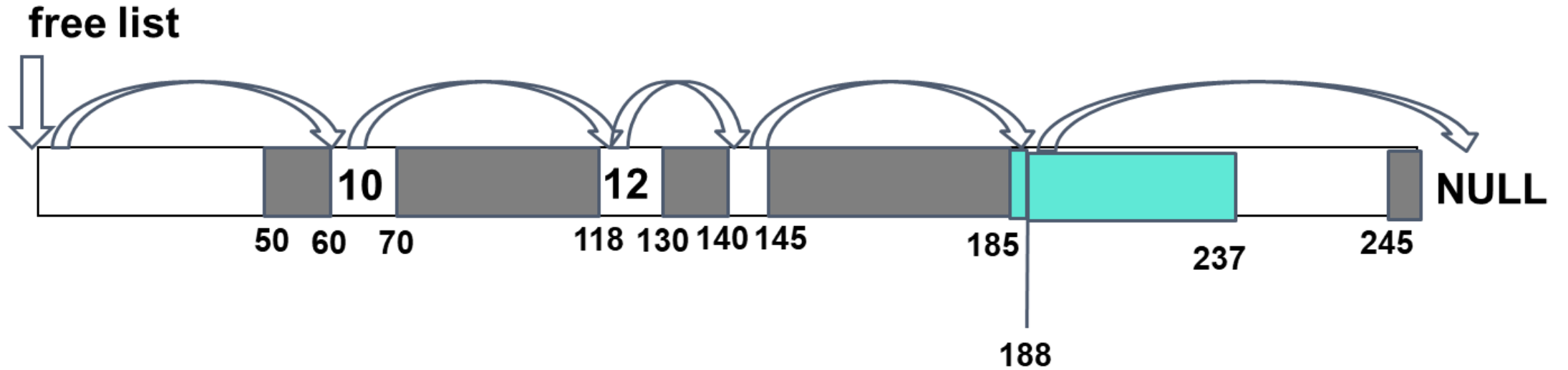
- הקצאה רק אחרי הבלוק הקודם שהוקצה עבורו מקום.
- מדידות מראות שמקבלים הרבה שברי זיכרון – עובד לאט.

Best-fit

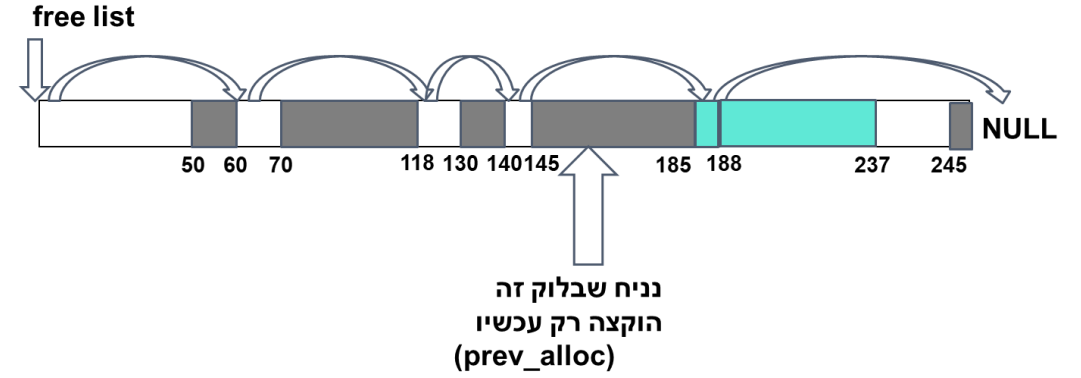
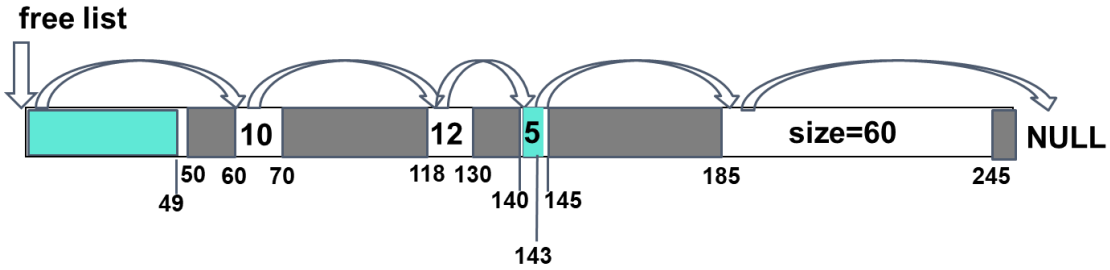


- הקצאה במקום הפנוי הקטן ביותר האפשרי.
- מקווים שזאת השיטה הכי חסכונית אך מדידות מראות שזהו לא המצב.

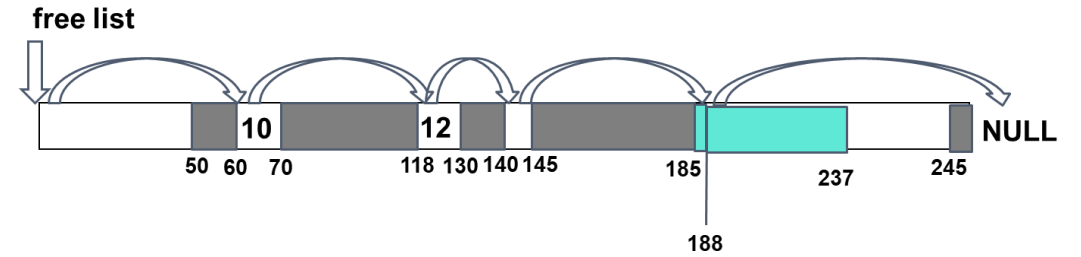
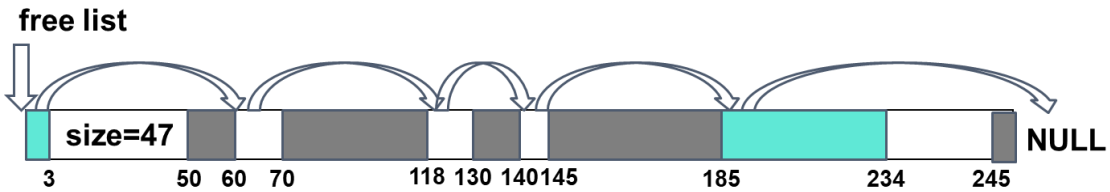
Worst-Fit

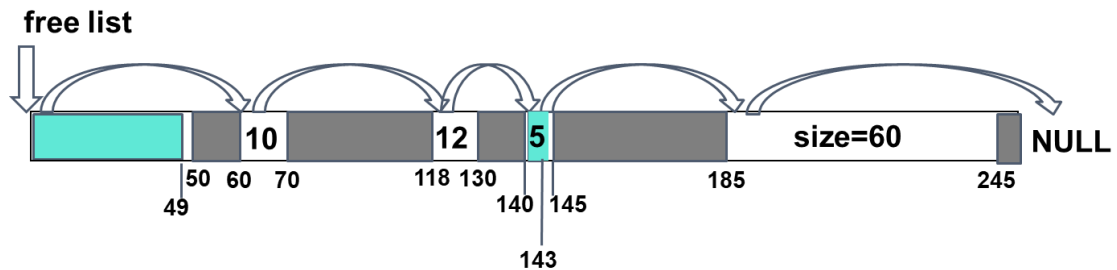


• הקצאה במקום הפנוי הגדול ביותר.

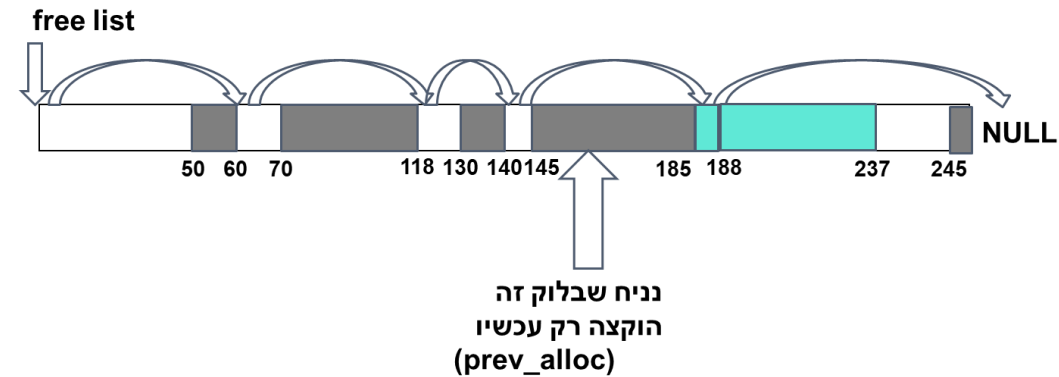


זהו את שיטת ההקצאה

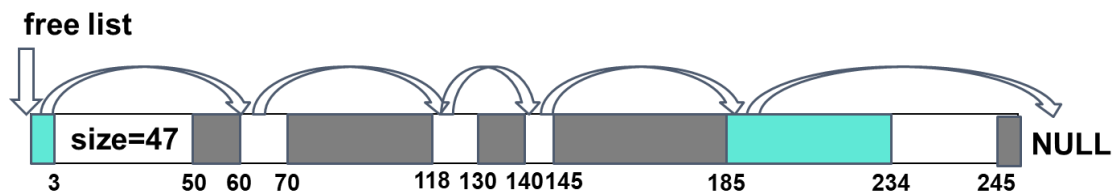




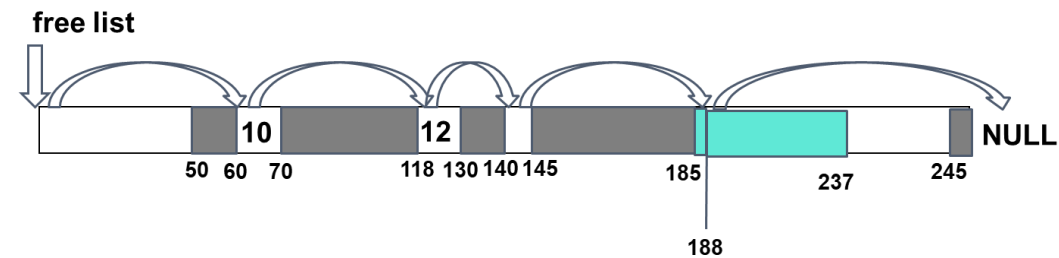
Best-Fit



Next-Fit



First-Fit



Worst-Fit

זיכרון וירטואלי - הקדמה

נפריד את הזיכרון הפיזי מהזיכרון הלוגי של התהליך.

כלומר: הכתובת שהתהליך מכיר היא לא הכתובת בה המידע נשמר.

הפרדנו? נרכיב!

נשתמש ב**מנגנון תרגום**:

- לצורך כך נדרש לתמיכת חומרה ומערכת ההפעלה.
- בצד החומרה: רכיב בשם **MMU – Memory Managemet Unit**.
- בצד מערכת ההפעלה: מבני נתונים לשמירת "טבלאות תרגום".

למה זיכרון וירטואלי?

מערכות בלי זיכרון וירטואלי סבלו מהבעיות הבאות:

1. באג בתהליך A דורס זיכרון בתהליך B או במערכת ההפעלה.

תהליך של משתמש אחר יכול לגשת לסיסמא שלי.

עם זיכרון וירטואלי: תהליך יכול לגשת רק לזיכרון שלו.

2. malloc משותף לכל התהליכים ולכן יש חורים בהקצאת הזיכרון שמשותפים לכולם.

עם זיכרון וירטואלי: אזורים לא רציפים בזיכרון פיזי נראים כמו זיכרון וירטואלי רציף.

למה זיכרון וירטואלי?

3. תהליך היה שוכח לשחרר זיכרון לפני שהסתיים (דליפת זיכרון).

עם זיכרון וירטואלי: מערכת ניהול הזיכרון יודעת מה שייך לכל תהליך ומה לשחרר.

4. התהליך לא ניגש לכל הזיכרון שלו כל הזמן – יש אזורים "קרים".

עם זיכרון וירטואלי:

בזיכרון הווירטואלי האזורים הקרים עוברים לדיסק בתהליך שנקרא **Page-out**.

ככה יוצא שהזיכרון הפיזי מתפנה לתהליכים אחרים.

לכן סכום הזיכרונות הווירטואליים גדול מה RAM.

תרגום כתובת וירטואלית לפיזית - דוגמא

נסתכל על שורת האתחול הבאה:

```
int *p = (int*)0x10452
```

גישה דרך *p היא לכתובת 0x10452 – כתובת וירטואלית.

הדוגמא הבאה תראה איך מתרגמים כתובת זו לכתובת פיזית.

המערכת שנדגים באמצעותה היא **קירוב** של מערכת זיכרון וירטואלית.

אי הדיוקים בדוגמא יפורטו בסוף המצגת.

תרגום כתובת וירטואלית לפיזית - דוגמא

קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B	
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

- לכל תהליך יש טבלת דפים פרטית.
- בזמן החלפת הקשר, המעבד עובר לטבלת הדפים הנכונה.

תרגום כתובת וירטואלית לפיזית - דוגמא

קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B	
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

• בניח שגודל דף הוא 4KB או 0x1000 בתים, ונראה:

1. Page-out כאשר הדף על הדיסק
2. Shared library או זיכרון משותף
3. Aliasing: כאשר 2 כתובות וירטואליות ממופות לאותו זיכרון הפיזי

תרגום כתובת וירטואלית לפיזית – דוגמא 1

	קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

1. תהליך A ניגש לכתובת הווירטואלית 0x10452
2. מחלקים בגודל הדף ומקבלים מס' דף 0x10 ו offset שהוא 0x452
3. כניסת הדף 0x10 מצביעה על המסגרת 0x20
4. מכפילים בגודל הדף ומוסיפים את ה-offset, ולכן הכתובת הפיזית היא:

תרגום כתובת וירטואלית לפיזית – דוגמא 1

קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B	
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

1. תהליך A ניגש לכתובת הווירטואלית 0x10452
2. מחלקים בגודל הדף ומקבלים מס' דף 0x10 ו offset שהוא 0x452
3. בניסת הדף 0x10 מצביעה על המסגרת 0x20
4. מכפילים בגודל הדף ומוסיפים את ה-offset, ולכן הכתובת הפיזית היא: 0x20452

תרגום כתובת וירטואלית לפיזית – דוגמא 2

קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B	
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

1. תהליך B ניגש לכתובת הווירטואלית 0x10452

מס' דף

offset

2. לכן הכתובת הפיזית היא:

תרגום כתובת וירטואלית לפיזית – דוגמא 2

קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B	
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

1. תהליך B ניגש לכתובת הווירטואלית 0x10452

מס' דף

offset

2. לכן הכתובת הפיזית היא: 0x30452

זיכרון משותף – דוגמא 1

קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B	
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

(אלא אם נאמר אחרת, הכתובות הן וירטואליות)

- תהליך B ניגש ל 0x12567 וזה יתורגם ל 0x21567
- תהליך A ניגש ל 0x11567 וזה יתורגם ל 0x21567

מה קרה כאן?

זיכרון משותף – דוגמא 1

קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B	
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

התהליכים משתפים ביניהם זיכרון. למה?

אם ספרייה מסוימת שמכילה יכולות משותפות כמו printf (פעולת הדפסה) שנטענת פעם אחת, אז כל התהליכים מצביעים לאותו הזיכרון הפיזי (של הספרייה) וחוסכים זיכרון.

זיכרון משותף – דוגמא 2

קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B	
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

- תהליך A כותב 5 לכתובת 0x10ABC שמתורגמת ל:
- תהליך A קורא מכתובת 0x13ABC שמתורגמת ל:
- הערך שיתקבל יהיה:

זיכרון משותף – דוגמא 2

קטע מטבלת דפים תהליך A		קטע מטבלת דפים תהליך B	
0x10	0x20 (valid, read/write)	0x10	0x30 (valid read/write)
0x11	0x21 (valid, read-only)	0x11	0x40 (valid read/write)
0x12	0xCD00 (invalid [paged out])	0x12	0x21 (valid read-only)
0x13	0x20 (valid, read/write)	0x13	0xFF000 (invalid [paged out])

- תהליך A כותב 5 לכתובת 0x10ABC שמתורגמת ל: 0x20ABC

- תהליך A קורא מכתובת 0x13ABC שמתורגמת ל: 0x20ABC

- הערך שיתקבל יהיה: 5

דפדוף

קטע מטבלת דפים תהליך B	
0x10	0x30 (valid read/write)
0x11	0x40 (valid read/write)
0x12	0x21 (valid read-only)
0x13	0xFF00 (invalid [paged out])

- תהליך B קורא מכתובת 0x13BAD
- דף 0x13 מסומן כלא תקין לכן מתקבל page fault
- מערכת ההפעלה מגלה שהדף נמצא בדיסק במקום 0xFF00
- הנתונים נקראים מהדיסק למסגרת 0x50 (למשל) בזיכרון הראשי
- האפליקציה ממשיכה מאותה הנקודה
- הפעם, היא מצליחה לקרוא מ- 0x13BAD שמתורגמת ל:

דפדוף

קטע מטבלת דפים תהליך B	
0x10	0x30 (valid read/write)
0x11	0x40 (valid read/write)
0x12	0x21 (valid read-only)
0x13	0xFF00 (invalid [paged out])

- תהליך B קורא מכתובת 0x13BAD
- דף 0x13 מסומן כלא תקין לכן מתקבל page fault
- מערכת ההפעלה מגלה שהדף נמצא בדיסק במקום 0xFF00
- הנתונים נקראים מהדיסק למסגרת 0x50 (למשל) בזיכרון הראשי
- האפליקציה ממשיכה מאותה הנקודה
- הפעם, היא מצליחה לקרוא מ- 0x13BAD שמתורגמת ל: 0x50BAD

אי דיוקים בדוגמאות

במציאות יש מספר רמות לטבלת התרגום אבל זה לא בחומר.

מי שרוצה להעשיר את הידע שלו, מוזמן לקרוא על זה ב:

- http://en.wikipedia.org/wiki/Page_table
- http://en.wikipedia.org/wiki/File:X86_Paging_4K.svg