

חשוב לזכור:

כשמגדירים משתנה בתור מצביע, הוא עדיין לא מצביע לשום מקום. לכן:

```
int *ip;  
*ip = 100;
```

היא שגיאה שמונעת הרצה (גורמת ל-crash).
דוגמא לשימוש נכון:

```
int *ip, x;  
ip = &x;  
*ip = 100;
```

אריתמטיקה עם מצביעים

```
float x=0, y=0, *flp, *flq;  
flp=&x;  
flq=&y;  
*flp = *flp + 10;  
++*flp;  
(*flp)++;  
flq = flp;  
++flq;
```



פעולות על משתנה אחד (כמו *
או ++), בניגוד לשניים (כמו +),
מבוצעות מימין לשמאל.

אריתמטיקה עם מצביעים

- לכל מצביע מציינים את סוג המשתנה שאליו הוא מצביע, בשביל שהמחשב ידע את כמות הזיכרון שבה מדובר.
- כשמגדילים את הערך של מצביע, הוא גדל ביחידה אחת בעלת גודל מתאים.
- ++ip מוסיף 2 בייטים לכתובת, אם מדובר במצביע ל-int.
- ++fp מוסיף 4 בייטים לכתובת, אם מדובר במצביע ל-float.

סיכום (ביניים)

- כל משתנה מקבל מקום בגודל מוגדר בזיכרון.
- משתנה מצביע מאחסן כתובת של משתנה. גם למצביע יש כתובת בזיכרון.
- הגדרת מצביע נעשית ע"י * לפני השם.
- אתחול מצביע בכתובת של משתנה מתבצע בעזרת פעולת & .
- ל- * תפקיד נוסף (אם אינו בשורת הגדרה) לציין בקשה להשתמש ב (או לשנות את) תוכן התא שכתובתו מאוחסנת במצביע.

Structures מבנים

שימושי כשמערך לא מתאים. למשל:

- ניהול כוח אדם:

שם עובד, מספר עובד, משכורת (יסוד ותוספות) - כלומר
גם היררכיות

- סימולציה של חלקיקים

מסה, אנרגיה, מטען, מיקום ...

מהו Struct ?

- ה- **struct** (קיצור של **structure**) הוא אוסף של משתנים מסוגים שונים (בניגוד למערך). הוא דוגמא של משתנה מסוג מופשט, **Abstract Data Type (ADT)**. הרכיבים במבנה נקראים חברים (**members**) או שדות (**fields**) והגישה אליהם היא לפי שם. הרכיבים במערך נקראים אלמנטים (**elements**) והגישה אליהם היא לפי מספר (**index**).

- הגדרת מבנה:

```
struct name_struct{ definition of variables};
```

דוגמא להגדרת מבנה

```
struct human {  
    char f_initial,l_initial;  
    int age;  
    float height; };  
/* Don't forget this semi colon!!!! */
```

- אפשר להשתמש בכל סוגי המשתנים, כולל מערכים, מצביעים, ומבנים.
- גודל המבנה שווה לסכום הגדלים של כל הרכיבים שלו: כאן $2*1+2+4=8$ בייטים.

הגדרת משתנה Structure

- אפשר עם רשימת משתנים אחרי ההגדרה:

```
struct album {  
    char *name, *artist;  
    int numTracks;  
} CD1, CD2, CD3;
```

כאן הגדרנו שלשה משתנים מסוג album. אבל אם ההגדרה היא מחוץ ל-main, אז הם משתנים גלובליים.

- אפשר במקום זה להגדיר בתוך main:

```
struct album CD1, CD2, CD3;
```

- או, עוד יותר טוב, מחוץ ל-main:

```
typedef struct album Album; ...
```

ואז בתוך main:

```
Album CD1, CD2, CD3;
```


פרויקט סיום

סימולציה של גופים תחת אינטראקציות של כוחות
כבידה וחשמל.

חומר עזר נתון:

- (1) דוגמאות של קבצי קלט
- (2) ספריית גרפיקה דו-מימדית
- (3) תכנית Matlab לגרפיקה תלת-מימדית

קבצים באתר: הוראות מלאות; דוגמא להצעת תכנון