**יום שני, חיים, 22.08.2022**

**זכרון**

**ספריית פונקציות חדשה <stdlib>**

1. משתנה מסוג פוינטר \*b.
2. פקודה חדשה: הקצאת זכרון בגודל כלשהו לבחירתנו שעליו יצביע הפוינטר.

B=malloc(sizeOf(typeOfVariable))

1. השמת ערך לפוינטר. b=100;.
2. סוף תכנית משחררים את הזיכרון שתפסנו. Free(b);.

יש 2 דרכים לאחסון זכרון:

* Stack- מחסנית (הזכרון הפרטי של התכנית).
* Heap- ערימה (ענן גלובאלי המכיל זכרון דינאמי עד שחרור), זכרון heap יותר איטי.

רק ברגע שמפעילים את התכנית הזכרון נכנס. בram נמצאים 2 סוגי הזכרון האלו

* משתנים המוגדרים באופן רגיל int a=10; נכנסים לstack.
* כאשר מבקשים הקצאת זכרון (אלקוצייה) בתכנית malloc() הזכרון הזה נכנס לheap עד שחרור free() והheap מחזיר את הזכרון.

פוינטר נמצא בשני סוגי הזכרון, בstack הוא תופס 8 בתים (שם נכנסת הכתובת).

בheap תופס מספר בתים לפי ההקצאה.

**חשוב:**

* כשמשחררים זכרון צריך לשחרר את תחילתה של כתובת לכן תמיד כדאי לעשות פוינטר העתק לתחילתו של הפוינטר שהקצנו לו זכרון ונקרא לו original.
* לא תמיד פקודת malloc() עובדת, לעתים אין מספיק מקום בזכרון להקצאה. לכן יש לבדוק בתנאי אם ההקצאה שעשינו לפוינטר==null אם כן להוציא הודעת שגיאה ולא להמשיך בתכנית.