

תכנון מונחה מ'נצ' (OOP)

Object Oriented Programming

פרדיגמה תכנון המשמש מ'נצ' /
אובייקטים
שלם כגוף ויכולת התחלה.

הפרדיגמה מספקת עמיתים מספר דרכים
עבר / ואפשר כגוף המושג באמצעות;
- כ'מ' (יצירת מ'נצ' של "קופסאות שחורות")
- הורשה (מחלקה Y יורשת ממחלקה X ומקבלת
ממנה את תכונות ופעולות ההורשה)
- עלמ'מ'פ'מ' (ע'פ'מ' אח'צ באובייקטים
הע'פ'מ' זכרון של מ'מ'פ'מ' אח'צ)

עקרונות הבסיסיים של OOP:

Data Abstraction (1
הפסג ג'מ'מ'

Encapsulation (2
כ'מ'מ'

C++

(3) מ'נצ'מ'

||

↓
(4) היררכיה (הוורטקס, סדר נורמלי)

הבדלים בין C ו C++

true/false

bool סוג (1)

```
bool isEven (int n){
    return n%2==0;
}
```

קריאה לקוד

חיסכון במקום -
1 byte

```
int val = ...;
```

```
if (isEven(val) == true)
```

Function overloading (2) סוגי פונקציות

```
C void swap_int(int*, int*);
void swap_float(float*, float*);
```

C++

```
void swap (int*, int*);
void swap (float*, float*);
```

{ }
{ }

```
int x=2, y=3;
```

```
float f1=2.5, f2=3.5;
```

```
swap(&x, &y);
```

```
swap(&f1, &f2);
```

① int add (float a, int b) { ... }

② int add (int a, float b) { ... }

int x=2, y=3, z=4;

float f=3.5;

✓ add(x, f);

✓ add(f, z);

add(x, y); X

✓ add((float)x, y);

```
void f(int)
{ ... }
```

```
void f(int,
int)
{ ... }
```

3) פונקציה עם פרמטרים ברירת

int sum(int=2, int=3);
הכרטיז
אין ארץ ההכרטיז

int sum(int x, int y)
{ return x+y; }

sum(); $\Rightarrow 5$
sum(4); $\Rightarrow 7$

sum(1,2); $\Rightarrow 3$
sum(, 4); //comp
Err

int sum(int , int , int = 0);

sum(2,4) ✓

sum(2,4,5) ✓

sum(2); } X

sum(0); }

Input > 13'3' Output (4

cin >>

cout <<

C

#include <stdio.h>

```
int a;  
float f;  
char str[80];  
fgets(str, 80, stdin);  
scanf("%d%f", &a, &f);  
printf("%d %s", a, str);
```

C++

#include <iostream>
using namespace std;

||

abc_xy

```
cin.getline(str, sizeof(str));  
cin >> a >> f; // 5 4.5  
cout << a << ' ' << str;  
5. abc_xy
```

standart input
stream

flow direction

cin → [] } program

ostream

standard output stream

cout <<
 כל ב"ר
 ש

```
cin >> str;  
cout << str;
```

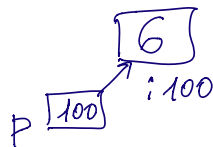
Ctf

$a = 5, f = 4.5000, str = abc \ x y \downarrow$

```
cout << "a=" << a << ", f="
      << f << ", str=" << str << endl;
```

malloc, free free / release
 new, delete allocate / create

C
 #include <stdlib.h>
 int *p;
 p = (int*) malloc(
 sizeof(int));
 if (!p) { }
 *p = 6;
 free(p);



int size, *p;
 scanf("%d", &size);
 p = (int*) malloc(size * sizeof(int));

for(int i=0; i<size; i++)
 cin >> p[i];

free(p);

C++
 int *p;
 p = new int;
 *p = 6;
 delete p;



cout << "Enter...";
 cin >> size;
 p = new int[size];

delete p;

reference אופנה/ת"חסוה

בהרצאה 2

! מחשבה ואובייקט על הקשור לסג