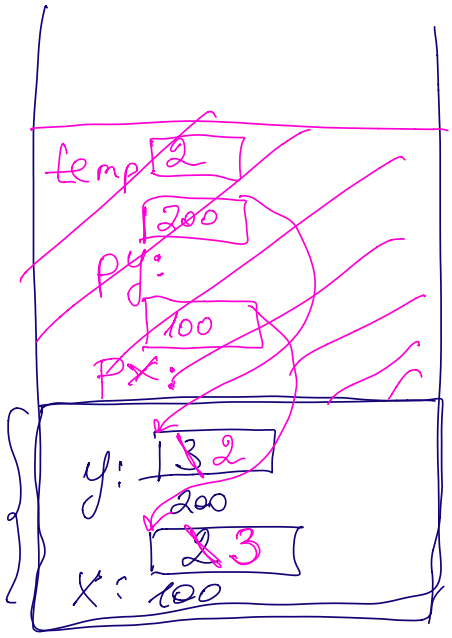
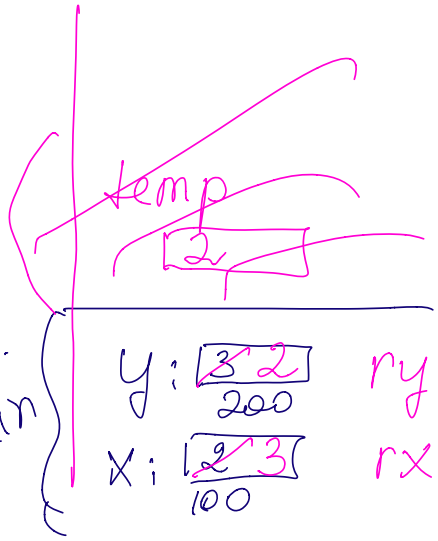
```
void swap(int * px  int * py)
{
  int temp = *px;
  *px = *py;
  *py = temp;
}
int main(){
   int x=2, y=3;
   swap(&x, &y);
}
```



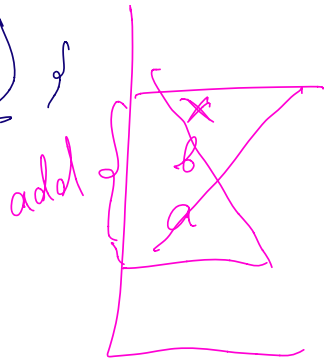
```
void swap(int& rx , int&ry)
{
   int temp=rx;
   rx= ry;
   ry = temp;
}
int main() {
   int x=2, y=3;
   swap( x , y );
}
```

int & rx=x



reference סוג נפוץ מאוד

```
int& add (int a, int b) {
    int x = a+b;
    return x;
}
```



או

```
int* add (int a, int b) {
    int x = a+b;
    return &x;
}
```

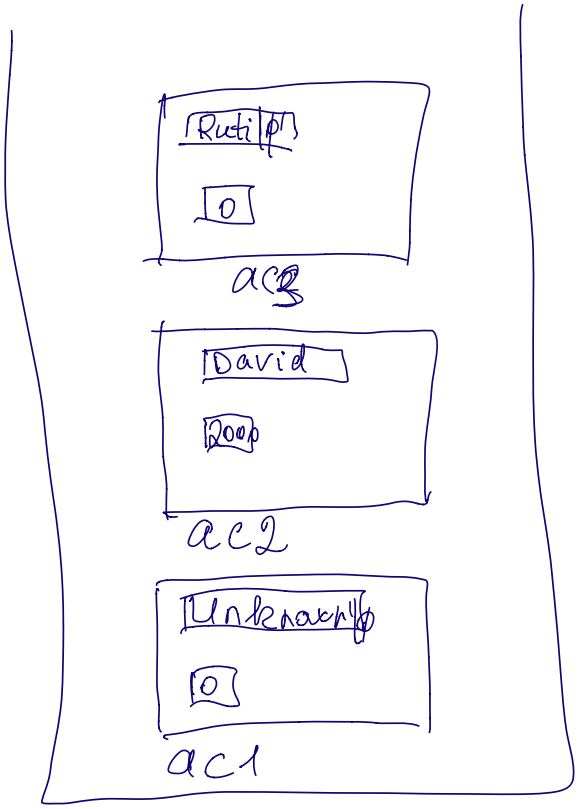
```
int add (int a, int b)
{ int x = a+b;
}  return x;
```

Ctor's מאוחר

```
class CAccount {
    char owner[30];
    float balance;
  public:
    void OpenAcc(....);
    void Show();
    .....
    CAccount(); //def ctor
    CAccount(char*, float=0);
    CAccount(){}
};

.cpp

CAccount::CAccount(){
    strcpy(owner, "Unknown");
    balance=0;
}
```

"מגלה"
הלכו טובה
מה-
Compiler

```
CAccount :: CAccount (char *name,
 float bal) {
    strcpy (owner, name);
    balance = bal;

}
int main() {
 CAccount ac1, ac2("David", 2000),
  ac3("Ruti");
  ac3.show();
```



stack

התהליך ביצירת אובייקט והפעלת
הבנאי :

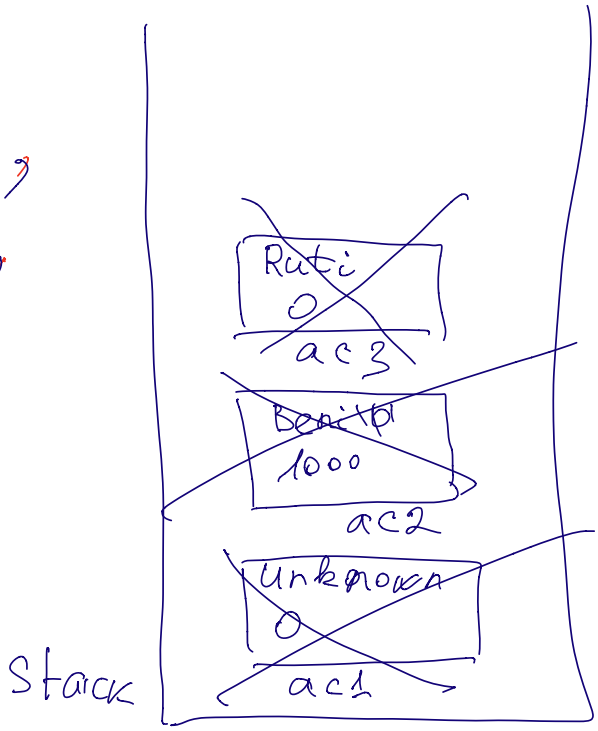
Object creation ⇒ memory Allocation ⇒ Ctor Activation

Destructors — Dtor (קיצור) — הורס

`~CAccount(){}` — מתנה מהקומפיילר

```
class CAccount{
    char owner[20];
    float balance;
public:
    ~CAccount( );    //dtor
    CAccount (...)            Ctor
};

CAccount :: ~CAccount() {
    cout << "I'm dtor ";
    show();
}
```

```
int main() {
  CAccount ac1,
  ac2("Beni", 1000),
  ac3("Ruti");

  return 0;
}
```



Dtor Activation ⇒ Memory Release Free ⇒ object Deletion

I'm dtor Ruti 0 ac3

I'm dtor Beni 1000 ac2

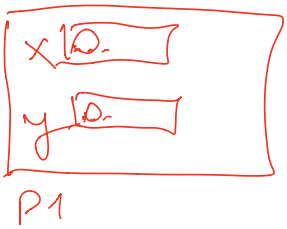
I'm dtor Unknown 0 ac1

דוגמה נוספת

```
class Point {
    int x, y;
  public:
        Point (int=0, int=0);
    :
};
```

Point.cpp

```
Point::Point (int X, int Y) : x(), y()
{  x = X;
   y = Y;
}
- - - -
int main(){
  Point p1, p2(5), p3(4,5);
```



```cpp
Point::Point(int X, int Y): x(X), y(Y)
{ }
```

מקרים בהם שימוש בשורת האיתחול חובה:

1) שדה/משתנה של מחלקה const

2) -||- reference

3) בהכלה, כאשר למחלקה המוכלת (הפנימית) אין def ctor

4) בהורשה, כאשר למחלקת האב אין def ctor