

18/07/23

מח"ל 11 - 20937 - 2202 (2)

עמוד 1: הקניין הקב"ט " Foo:Bar "

הסבר על התחסינות אפלטוניסטית:

המחלק Bar נמצא על Foo. כמ שמעבר המעברים 33-34
המפיק, ב- ++C הפינקציות אינן ויזואליות כדורית אחד ^{לפי} בקוד
הנפמן הנקרא המפורש - זכטל void תהיה ויזואלית, ובמחלקה
Bar המעריכי הלו נרמזת, מה שמאפשר לעתים אפיה כך שמצאן
הצב תופסל העיסה א Bar ^(הבציות) עם זכטל, בקוד הנפמן הפינקציה
זכטל מכסלת כחלק מהבנה א Foo. מכיוון שהצב בנייה
נקרא קודים בעי (האבא), מה שמאפשר הווא דיוקא המימוש
א זכטל בתוך Foo. אכן, המבנית ככלל לא עושה
שימוש במעריכי הקבצ שנימתי ב- Bar, שאם יתקבלו
שבאבא (כחלק מהבניה), כמו שמעבר בסוף צל המצגת
א יחידה ב שגלחה און.

שאלה ב: הבעיה פקודה היא free double. $Point\ p_2 = p_1$
 נניח שיש לנו המשתנה p_1 המכיל כתובת זיכרון של $Point$
 באותו זמן, יש לנו קובץ $other$ (const Point & other) $Point$
 שבתוכו אנו - p_2 מחזיקה זיכרון נוסף את ערך הכתובת
 של $coord$ - בשימוש p_1 ישירות לשם
 $coord$ של p_2 . הבעיה היא שבתוך הריצה מה-main
 מתחיל מחיקת המימון של p_1 ושל p_2 .
 שם נכנסו שתיים מביטות אלו מתים כיבדו שניהם פעמים -
 הפונקציה free מזהה free double כי אין זה אלא אותו הפוסט (תמיד).
 (זה לא להשיג באפריקה) 'לא הוציא לביצוע' הוראה
 עתידית. ליתרון, הבא (עדיף) את הקוד כן שיתאפשר
 'על הדרך' מתקין; כמו שמוסרי המודלים - גם במדריך:

```
Point (const Point & other) { // בנייה
    - coord = new int[2];
    - coord[0] = other.coord[0];
    - coord[1] = other.coord[1];
}
```

// אפוא

```
Point & operator = (const Point & other) {
    if (this != &other) {
        delete[] coord;
        coord = new int[2];
        coord[0] = other.coord[0];
        coord[1] = other.coord[1];
    }
    return *this;
}
```

כמו בעמוד 30 במדריך