**אמצעי תכנות של משתני (פרמטר בעיקר) const**

**העברה by value נועדה להבטיח מניעת "תופעות לוואי" (side effects) משתנה המקור לא עובר שינוי רק בגלל שהועבר כפרמטר.**

**הגנה כזו לא קיימת ב-Java ו-C# (שם המופע עשוי לעבור שינוי בכל מקרה).**

**העברהby value כרוכה בשכפולים, עשוי להיות יקר, פתרון אפשרי הוא להעביר by reference ו-const.**

**משתנה const מקבל ערך בבנאי והקומפיילר לא יקמפל קוד המשנה ישירות אחר כך (ניתן לעקיפה ע"י פוינטרים ו-casting).**

**קוד על משתני const יעבור קומפילציה רק אם:**

* **מדובר בקוד קורא של שדות,**
* **מריץ רק מתודות const.**
* **מתודות const נבדקות בזמן קומפילציה שאינן כוללות קוד שמשנה את המופע.**
* **על פי רוב, אם אנחנו מממשים מתודה שאינה משנה את המופע, ולא סביר שזה ישתנה בעתיד כדאי להגדיר אותו const.**