1

בעיות במצביעים רגילים:

-ייתכן שהפוינטר מצביע לנאל והמפתח שכח לבדוק. (בהגדרת המצביע בד"כ נאתחל לנאל).

-מצביע שלא אותחל לנאל

-מצביע שהוקצה הקצאה דינמית ולא שוחרר בסוף התוכנית

-שלואו קופי בהשמה בין מצביעים (בלי אופרטור השמה)

**מצביעים חכמים-**

**מאפיינים:**

-בניה והריסה: נוכל לקבוע מה קורה כשהמצביע נוצר.

-העתקות בין מצביעים: שליטה על מה שיתבצע (דיפ קופי וכו)

-התייחסות: מה קורה כשמתייחסים לאובייקט

Graphical user interface, text, application

Description automatically generated**מימוש**:

אופרטור השמה בשביל דיפ קופי

אופרטור אוברלודינג לאופרטורים שמשתמשים בפוינטרים

Graphical user interface, text, application

Description automatically generated

במקרה זה 3 פוינטרים מצביעים על אותו "רטיונל".

כדי לפתור בעיה זו, ברגע שניצור מצביע חדש שמצביע לאותו אובייקט, נדאג שהישן יצביע לנאל. נבצע **העברת בעלות**.

Text

Description automatically generated

**"תופעות לוואי" של פוינטרים מסוג זה:**

1.

Graphical user interface, text, application, email

Description automatically generated

לא קיבלנו רפרנס

ולכן יופעל קופי קונסטרקטור שידרוס את הערך של הפוינטר הישן

לכן כאן, יודפס 0, לאחר שהערך אופס בגלל ההעתקה

**בד"כ בהעברת פוינטר חכם נרצה להשתמש ברפרנס**

2.