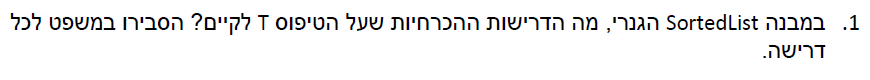
{יבש 3}

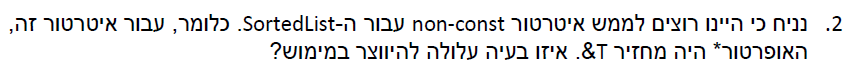
מגישים:

אראג מחיטריאן: 315031203

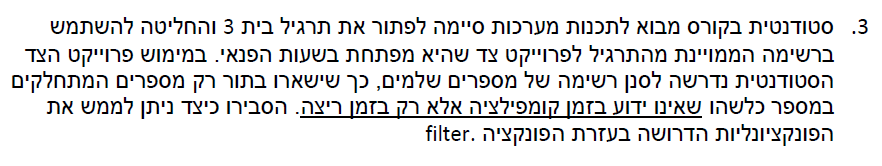
חוסיין עזקי: 213382104

**תשובה:** הדרישות על הטיפוס T בהקשר של המימוש שלנו למחלקה SortedList<T> הן:

1. **בנאי העתקה:** כדי שנוכל לבצע העתקה של הערך ל T Data שמתקבל לתןך חולייה ברשימה ששומרת את הערך ללא תלות בזיכרון המשתמש. כפי שהתבקשנו. במילים אחרות השמירה של ערכי T חייבת להיות by value ולא by reference.
2. **אופרטור בינרי > :** לטעמי מיון בתוך הרשימה הכרחית היכולת לדעת מה הסדר בין שני אברים t1,t2 מטיפוס T.
3. **הורס:** מאחר והמימוש שלנו שומר עותקים של T , בהורס של כל חולייה צריך לקרוא להורס של T (זה קורה ברקע ולא במפורש כמו לקרוא להורס של אובייקט שמצביעים עליו) חייב של T יהיה הורס כי אחרת עלולות להיות זליגות זכרון עקב השימוש ברשימה שמימשנו שהן תלויות בכלל בטיפוס T.



**תשובה:** הבעיה שעלולה לקרות היא שהמשתמש יכול לשנות את הערכי T במבנה עצמו כרצונו ולהרוס את המיון שהמבנה צריך לשמור עליו. ואין שום דרך למבנה לדעת שאחד מערכיו השתנה בצורה שמצריכה סידור מחדש. למשל עבור T שהוא בפועל int , אם הרשימה המקורית היא {3,2,1}, אם נשנה את ה 2 ל 5, היה מצופה שהרשימה תהפוך להיות {5,3,1} , בפועל זה לא מה שיקרה ונקבל {3,5,1}.



**תשובה:** הסטודנטית תצטרך להשתמש ב SortedList<int> ולשלוח לפונקצייה filter פרדיקט עם אופרטור () שמקבל Int ושמחזיר true אם הקלט לאופרטור מתחלק בערך k שמאותחל איתו אותו פרדיקט בזמן ריצה ו false אחרת.

דרך המימוש ממש דומה לפנקטור שמימשנו ברטוב לבדיקת TaskType.