

שאלות חלק יבש :

1. טיפוס T חייב להיות CLASS, וקיים את הדברים הבאים
 - a. Copy-Constructor כיוון שניתן לבנות רשימה ריקה ולכן נצטרך ליצור טיפוס T שמצביע למקום ריק (נתון שאין להניח על בנאי ריק)
 - b. Distroctur - אחרת לא ניתן למחוק איברים מהרשימה
 - c. Operator > - אחרת לא ניתן לבצע השוואה בין איברי הרשימה ובכך ליצור רשימה ממויינת (בעזרת אופרטור הזה ניתן לבצע השוואה הדומה לאופרטור ==)
 - d. Operator = - בהעתקה בין 2 רשימות אנחנו משווים, אך לפי ההנחיות של הקורס אופרטור השמה יהיה מכיוון שאם יש לפחות אחד מהBIG THREE יהיה גם הנוסף
2. הבעיה Iterator-non const זה כאשר היינו ממשים את האופרטור * המשתמש היה מקבל את הרפרנס לערך שנמצא ברשימה שה Iterator מצביע עליו, ולכן המשתמש יפגע במימוש של הרשימה הממוינת - המימוש עובד כך שהמשתמש יכול לשלוט על איברי הרשימה רק ע"י הפעולות insert, remove ולכן שינוי יאפשר לו שינוי בערכי הרשימה => שינוי בערך התא עלול להוביל לפגיעה בסדר הרשימה ובכך לא תהיה ממוינת.
3. 2 הדרכים להעברת פרדיקט זה ע"י :
 - a. לייצר פונקציה ולהעביר את הארגומנט עצמו
 - b. יצירה של מחלקה עם פונקציה אופרטור (). ולהעביר את הפרדיקט מהמחלקה כארגומנט
 - c. לסיכום: אנחנו לא צריכים 2 דרכים שונות כדי ליישם את העברת הפרדיקט, את 2 הדרכים אנחנו ממשים באופרטור () => דרך אחד