

3.1.1

ראשית, לT צריך להיות בנאי העתקה (שמעתיק העתקה לא רדודה) על מנת שבהכנסת איבר חדש לרשימה (במתודה INSERT) הערך החדש שיוכנס יכיל כתובת משל עצמו ולא ימשיך להיות מקושר לכתובת של הפרמטר הועבר למתודה (כדי למנוע דריסות של ערכים ומחיקות כפולות).

כמו כן, לT צריך להיות מימוש של אופרטור < על מנת שנוכל לבצע את ההשוואה בין שני ערכים ולהכניס את הערכים לרשימה באופן ממויין (בהתאם להגדרות הטיפוס T).

בנוסף, אם מדובר בטיפוס נתונים מורכב (שכולל בתוכו הקצאה דינמית של זיכרון) רצוי יהיה destructor שמשחרר את הזיכרון כראוי.

3.1.2

אילו היה איטרטור לא קבוע שמחזיר T& אז הלקוח של הממשק (SortedList) יוכל לשנות את ערכי הרשימה (לדרוס, לשנות וכו') מה שיפגע כמובן בסדר הממויין של הרשימה.

3.1.3

הפונקצייה filter בעצם מקבלת object function, כלומר פונקציה גנרית ממבנה מסויים ולכן ניתן להעביר בה כל פונקצייה בעלת אותה חתימה (מקבלת T ומחזירה bool במקרה של התרגיל שלנו). לכן ניתן לקרוא לפונקצייה filter על רשימת שלמים ולהעביר אליה פונקציה שבודקת עבור כל מספר האם הוא מתחלק במספר כלשהו. מכיוון שהחתימה היא גנרית (בזכות הobject function) אז ניתן להעביר שם כל פרמטר מהקלט וזה יענה על הצורך.



מימ לתחרות מימז: