

## חלק יבש

1. על הטיפוס T לקיים שלוש פונקציות בשביל השימוש ב-SortedList:  
א. Copy Constructor – לצורך העתקת המידע המתקבל לתוך Node חדש, ובפונקציות שמחזירות את התוכן של T.  
ב. operator< - לצורך מיון הרשימה לפי "גודל" המידע, בהכנסת איבר חדש.  
ג. Destructor – על מנת לשחרר את המידע בסוף השימוש.

אחרת, הקוד לא יעבוד כראוי.

2. המידע ששמור בכל חוליה ברשימה היה נתון לשינוי על ידי המשתמש. כלומר, יכול היה להיווצר מצב שבו יצרנו רשימה ממוינת, ואז המשתמש החליט לשנות את הערך ששמור בערך החזרה של האיטרטור, כלומר – ערך שמאוחסן באחת החוליות ברשימה, מה שהיה משנה עבורנו את הערך ששמור ברשימה ובסבירות גבוהה פוגע במיונה.

3. דרך ראשונה, כמו שהשתמשנו בפועל:

```
template <class T>
template <class Condition>
SortedList<T> SortedList<T>::filter(Condition condition) const
```

הפונקציה filter מקבלת כארגומנט את הקלאס המסנן.

דרך שנייה היא להשתמש במצביע לפונקציה, לדוגמה:

```
bool *condition (T);
```

יכולנו להסתפק רק במימוש אחד מכיוון ש- C++ יודעת להתייחס אל operator() כאל פונקציה.