

## Dry Part

1. הטיפוס T צריך לתמוך באופרטור < כיוון שהרשימה היא רשימה ממוינת גנרית שמכילה אובייקטים מסוג T לא ידוע על מנת שנוכל לקבוע את אופן המיון של הרשימה, בנוסף על T לתמוך בבנאי ההעתקה כדי שהוא יוכל ליצור עותקים חדשים של הטיפוס ברשימה חדשה ולא ישתמש באובייקט המקורי ששייך לרשימה הקיימת.
2. אם האיטרטור יהיה const non const ונפעיל אותו על מחסנית שהיא const הבעיה היא בכך שאופרטור \* יחזיר רפרנס שהוא לא קבוע מה שיאפשר שינוי של איברי הרשימה וזה יצור בעיה בזמן המימוש גם אם נגדיר את המתודות של האיטרטור להיות const
3. על מנת לפתור את הבעיה ניתן להגדיר ביטוי למבדא שבודק את החלוקה במספר num שמתקבל כפרמטר מהפונקציה הקדומה ומחזיר ערך בוליאני עבור כל איבר ברשימה a שמתחלק ב num באופן הבא:  

```
[num](int a){return num==0 ? throw("Can not divide by zero"); : a%num == 0;}
```

  
הפונקציה filter תקבל את הביטוי ובזמן ריצה תבנה רשימה ממוינת של כל האיבר שמתחלק ב num אם num==0 זורקים חריגה של חלוקה ב-0.



