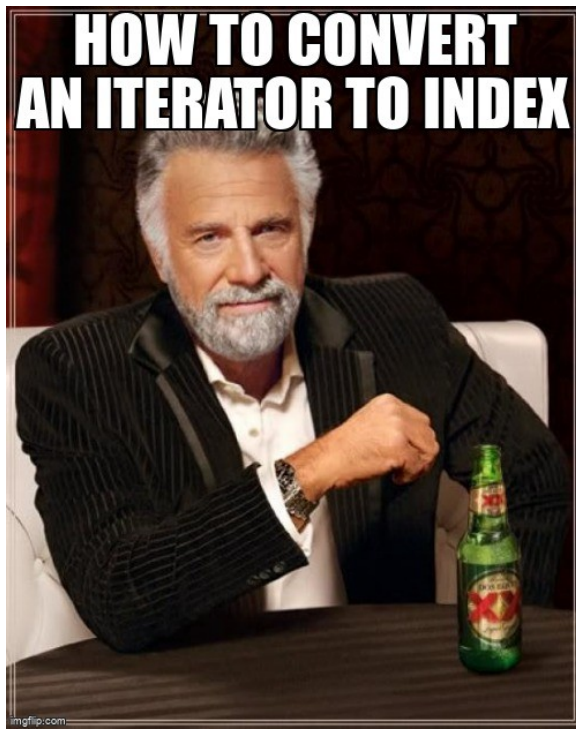


# Семинар 12 по ООП

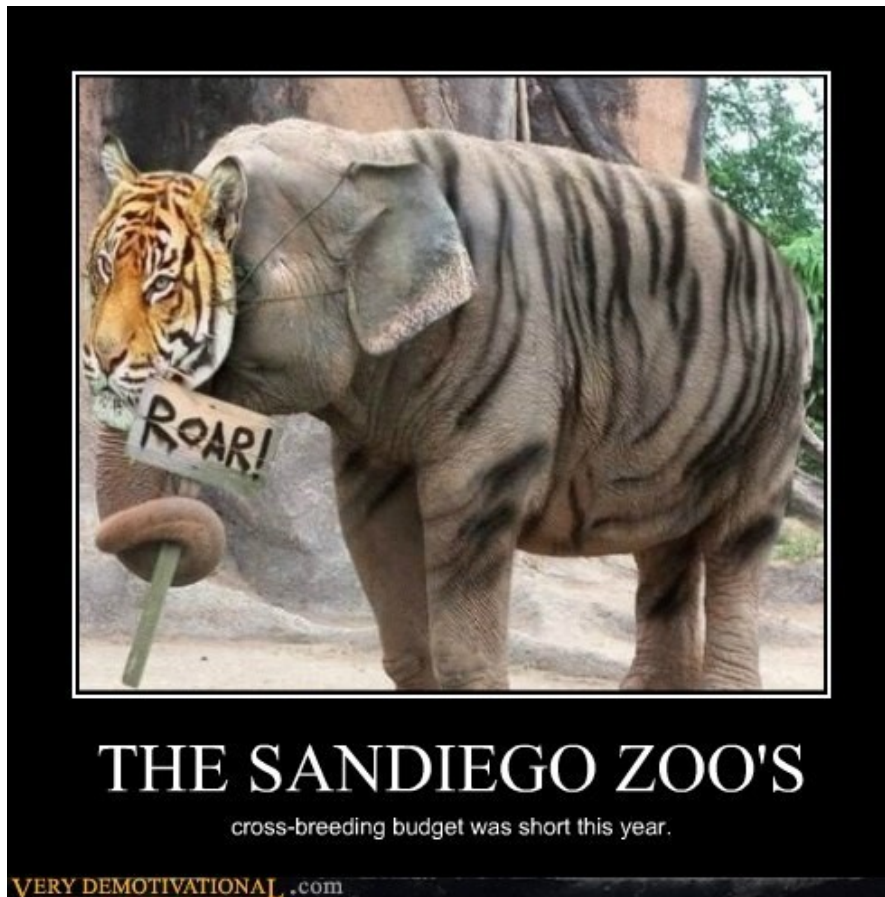
Хетерогенни контейнери, Итератори



# Хетерогенни контейнери

- Това са контейнери, в които можете да вкарате неща от различен вид.
- Пример: Зоопарк – там различни видове животни са набутани на малка площ, но в различни клетки, като единственото, което ги обединява е фактът, че са животни
- В програмирането може да вкараме указатели към инстанции на различни класове, които имат общ родител в общ масив от указатели от типа на общия родител

# Примерът със Зоопарка на код



# Какво видяхме

- Съхраняваме инстанции на различни животни в един зоопарк.
- Наблягам на това, че масивът е от динамичен масив от указатели, а не от инстанции!!!
- Методите, които ще имат различно поведение са виртуални!!! Така включваме динамичното свързване!

# Подчертавам

- Трябва да направите нужните виртуални конструктори за копиране и виртуални деструктори (при нужда)
- Аз не съм ги реализирал защото това не е фокусът на днешното упражнение.

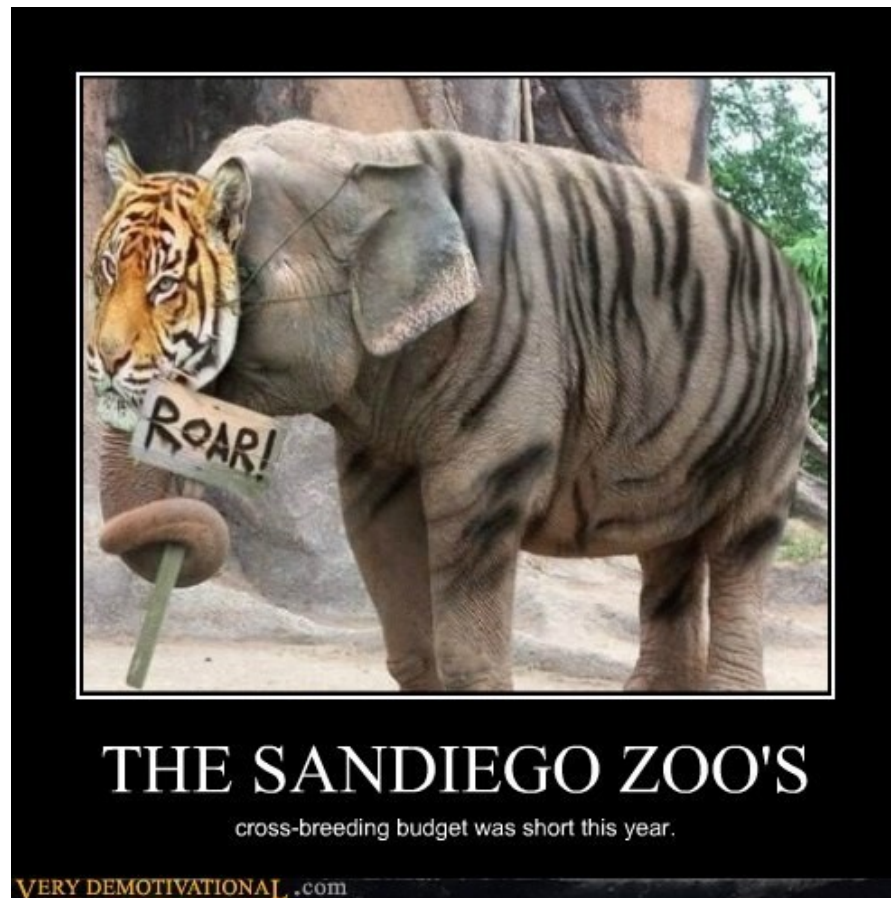
# Итератори

- Указатели на стероиди (буквално)
- Защото явно е добра идея да абстхарихаме указател.
- Идват с допълнителни блягинки като Range based loops

# Що е итератор

- От името му става ясно, че ще се използва за обхождане на някаква структура.
- Клас, в чиято `private` част има указател към елемент, от който е съставена колекцията, а в `public` частта има оператори за модификация на този указател.

# Примерът със зоопарка с итератор





# Design Pattern-и

- За да не преоткриваме топлата вода.
- Те представляват утвърдили се шаблони за ООП дизайн измислени и описани от умните глави, които са настъпали някакво крайно множество мотики (крайно не значи малко) многократно с цел да не бъдат настъпвани повече от хората.
- Примери: Command Pattern, Builder Pattern, Factory Pattern, Observer Pattern, Singleton Pattern и мн. Други
- Тези неща ще са ви полезни за проектите, защото докато пишете, вие е много вероятно да настъпите някоя друга мотика от гореспоменатото крайно множество.

# Пример (Command Pattern)



# Задачи

- Да се реализира линеен едносвързан списък с итератор
- Да се реализира следната йерархия. Имаме различни видове фигури. Кръг, квадрат, триъгълник. Всяка фигура предоставя интерфейс за намиране на обиколка и лице. Да се напише хетерогенен контейнер за фигури, който има итератор
- Да се реализира клас вектор с итератор
- Ако има време Observer Pattern (ако няма вижте го сами не е труден)