

iOS Development

Set up

About me

Vadim Zhuk
iOS Engineer





ABOUT YOU

Введение в Swift

- Типы данных
- Кортежи
- Опциональные типы
- Циклы и условные операторы
- Массивы, множества, словари
- Функции
- Замыкания
- Перечисления
- Основы объектно-ориентированного программирования
- Структуры и классы
- Инициализаторы и деинициализаторы
- Расширения и протоколы
- Обработка ошибок и отладка кода
- Основные алгоритмы
- Сложность алгоритмов
- Протоколы
- Протокол-ориентированное программирование

Введение в iOS

- Системы управления версиями
- Библиотеки
- Введение в паттерны программирования
- Основы работы с интерфейсом
- Таблицы
- Анимации
- Многопоточность
- Базы данных
- Управление памятью
- Работа с сетью
- Защита проекта

SWIFT

1. Иntro
2. Типы данных
3. Struct, Class, Enum, Tuple
4. String
5. Опциональные типы
6. Массивы, множества, словари
7. Циклы и условные операторы
8. Basic algorithms: sort, find
9. Git, IDE, app lifecycle
10. Interface builder, AutoLayout
11. Interface: Storyboards vs Xib
12. UIViewController
13. Protocols, Delegation pattern
14. Extensions

15. TEST

iOS SDK

1. UINavigationController
2. UIScrollView, UITableView
3. UITableViewCell
4. Architecture patterns, MVC, SOLID principles
5. Libraries, frameworks
6. Core Animation / Animations
7. UICollectionView
8. Network
9. Concurrency, Multithreading
10. Concurrency, Multithreading 2
11. Gestures
12. Memory
13. Debugging / Reverse Engineering
14. KVO
15. DataBases
16. CoreGraphics
17. Notifications
18. *CameraKit*
19. *Localization*
20. *MapKit, CoreLocation*
21. *User Notifications*
22. *SwiftUI*
23. **FINAL PITCHES**



TOOLS

Xcode

Terminal

Telegram/Slack

Zoom

GitHub



Git

Система контроля версий кода при разработке.



Swift

Programming language



SDK

iOS/iPadOS, tvOS, watchOS

**Apple developer account
GitHub
Telegram**

IDE

Integrated Development Environment

ООП

Объектно-ориентированное программирование — методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является **экземпляром определённого класса**, а классы образуют иерархию наследования.

CLASSIFICATION OF ANIMALS

This is the grouping together of animals with similar characteristics. Animals can be classed as either vertebrates or invertebrates.

ANIMALS

VERTEBRATES

These are animals that have a backbone.



Reptiles

Have dry scaly skin.
Lay eggs on dry land.
Are cold blooded.
(Snake, Crocodile)



Fish

Have scales on their bodies.
Have gills for breathing.
Are cold blooded.
(Shark, Tuna)



Amphibians

Have moist slimy skin.
Lay eggs in water.
Are cold blooded.
(Frog, Newt)



Birds

Have feathers and wings.
Have beaks and lay eggs.
Are warm blooded.
(Wren, Swan)



Mammals

Have fur or hair.
Feed young on milk.
Are warm blooded.
(Cow, Human)

INVERTEBRATES

These are animals that do not have a backbone.



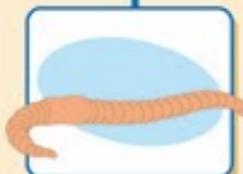
Protozoa

Single cell organisms
all microscopic.



Flatworms

Simple and soft bodied.
(Tape worm, Flukes)



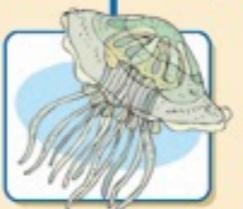
Annelid Worms

Segmented bodies.
(Earthworm, Leech)



Echinoderms

Spiny sea creatures.
(Starfish, Sea urchin)



Coelenterates

Soft bodies, stinging cells.
(Jellyfish, Sea anemone)



Molluscs

Soft bodied, most have shells.
(Snails, Limpet)



Arachnids

Eight legs, two body
parts, no antennae.
(Spider, Scorpion)



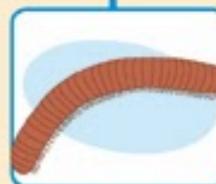
Crustaceans

Mostly sea creatures.
Many legs and two
sets of antennae.
(Crab, Lobster)



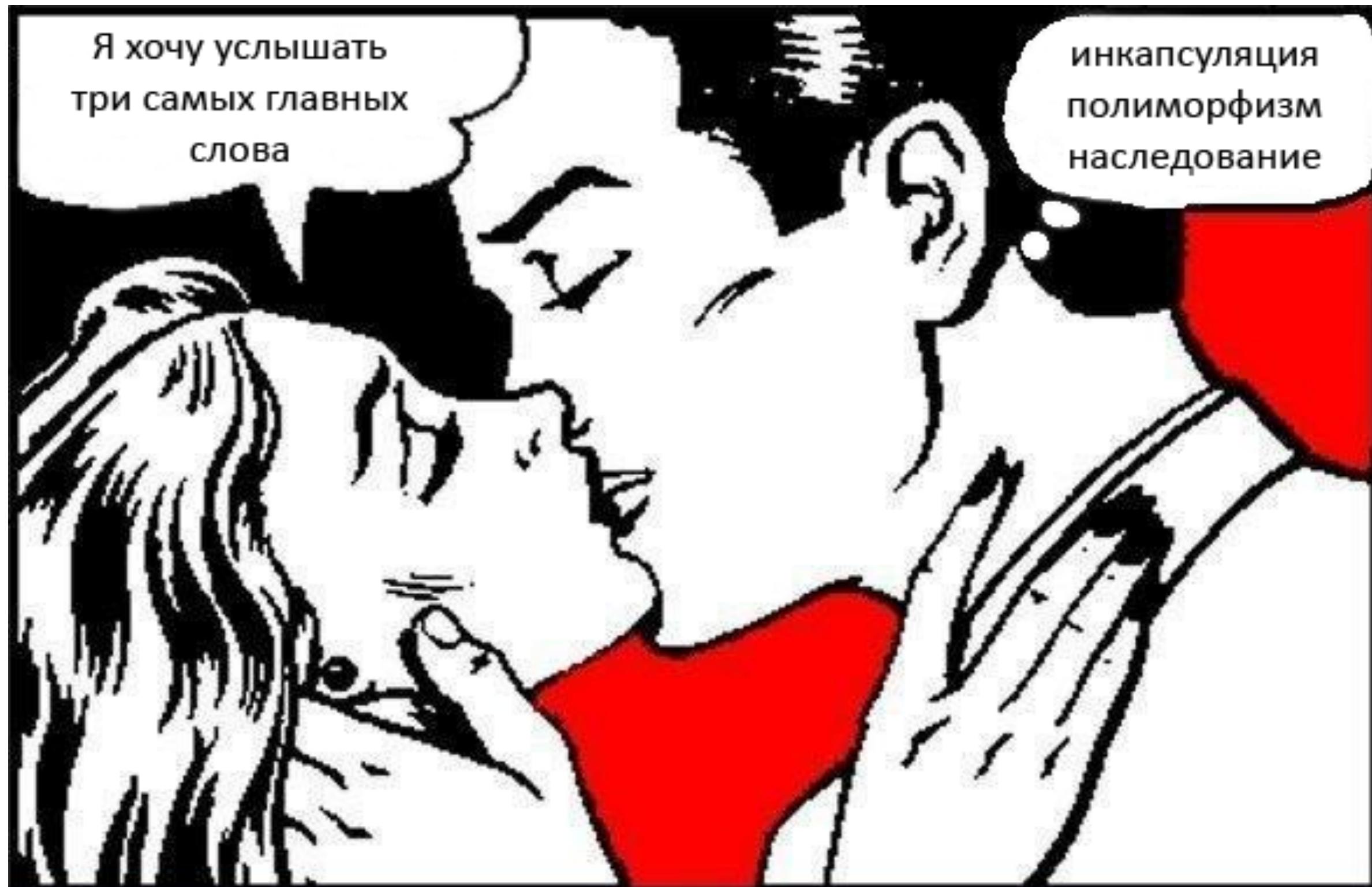
Insects

Wings, six legs, three
body parts, one pair
of antennae.
(Bee, Ladybird)



Myriapods

Many legs and
body segments.
(Centipede, Millipede)



Я хочу услышать
три самых главных
слова

инкапсуляция
полиморфизм
наследование

Абстрагирование – это способ выделить набор значимых характеристик объекта, исключая из рассмотрения незначимые.

Абстракция – это набор всех таких характеристик. Отделение концепции от ее экземпляра

Наследование – это механизм системы, который позволяет, наследовать одними классами свойства и поведение других классов для дальнейшего расширения или модификации.

Полиморфизм — свойство системы, позволяющее иметь множество реализаций одного интерфейса.

Ничего не понятно.

Инкапсуляция – это контроль доступа к полям и методам объекта. Под контролем доступа подразумевается не только можно/нельзя, но и различные валидации, подгрузки, вычисления и прочее динамическое поведение.

Wishes

PROGRAMMER IS NOT JUST DEVELOPER, BUT AN ENGINEER
GET GOOD AT PROBLEM-SOLVING
GET GOOD AT RESEARCHING
MAKE GOOD HABITS EARLY
FIND A COMMUNITY
MISE A PLACE (FR.) - SET UP
YOU DON'T NEED TO KNOW EVERYTHING
LEARN OTHER LANGUAGES
FIND SIDE INCOMES OF IDEAS
DON'T COMPARE YOURSELF TO OTHERS