**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики и информатики**

**Кафедра информационных систем управления**

**Отчет**

**По технологиям программирования**

Выполнил студент группы №12

*Шишлянников Иван Викторович*

**Минск 2020**

**Вариант 3.**

1. Напишите программу расчета и вывода первых десяти факториалов.
2. Дано двухзначное число.

Определить:

а) входят ли в него цифры 3 и 7;

б) входят ли в него цифры ( 4 и 8) или цифра 9.

5. В заданной строке посчитать количество слов, содержащих только строчные русские буквы. Вывести список таких слов, отсортированный в обратном порядке.

Разделителями слов считаются пробелы. Вывести.

## ЗАДАНИЕ 1. ЗНАКОМСТВО СО СРЕДОЙ ПРОГРАММИРОВАНИЯ XCODE»

**Цель задачи:** Изучить среду Xcode и научиться создавать проект iOS.

**Задачи**:

Создать проект iOS на языке swift.

Научиться запускать iOS Simulator.

Подключить свой Apple ID в настройках Xcode.

Включить результаты выполнения в отчет лабораторной работы 6. **Ход работы:**

*1.1 УСТАНОВКА СРЕДЫ ПРОГРАММИРОВАНИЯ XCODE*

Для того, чтобы установить среду разработки Xcode, открываем приложение App Store. В поиске набираем Xcode и открываем страницу приложения (рис.1). В левом верхнем углу, под иконкой приложения, нажимаем кнопку “Загрузить”. Ждем окончания загрузки.

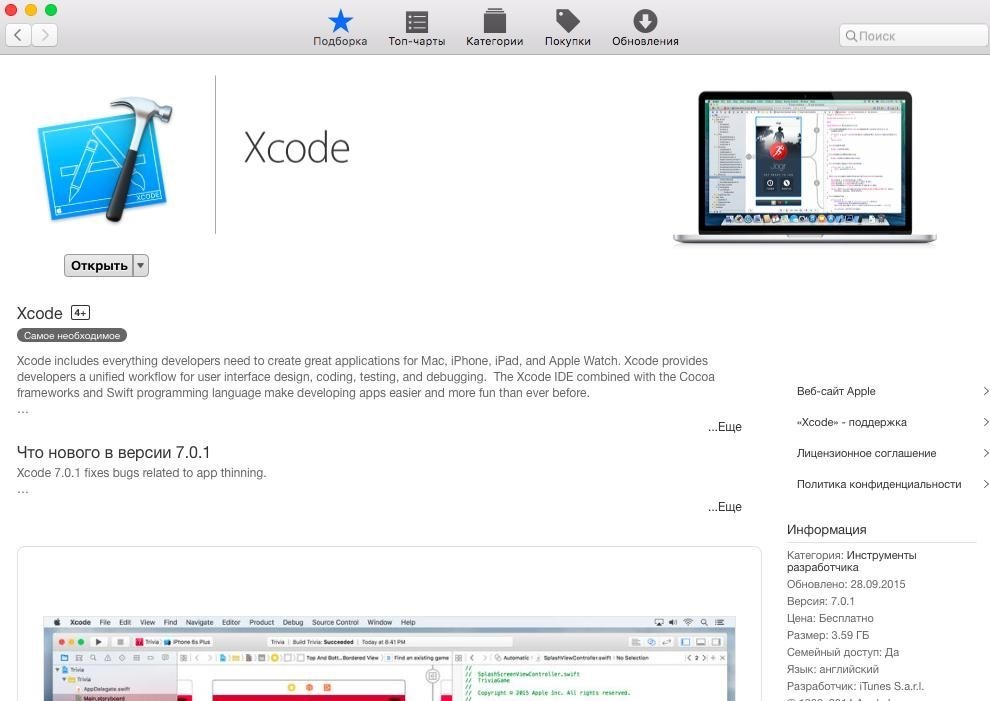


Рис.1. Страница приложения Xcode

*1.2 ЗАПУСК СРЕДЫ ПРОГРАММИРОВАНИЯ XCODE*

После завершения загрузки переходим в папку “Программы” на вашем компьютере Mac, либо нажимаем клавишу F4. Находим приложение Xcode и запускаем его (рис 2).

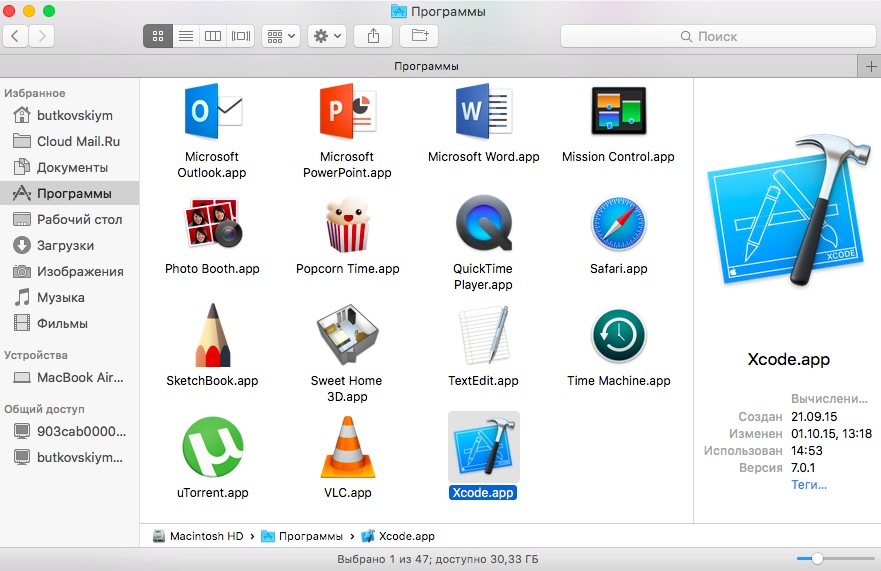


Рис. 2. Запуск программы

Появляется окно запуска (рис. 3), где мы можем:

* Быстро открыть недавние проекты (таблица в правом углу), либо открыть уже существующий проект;
* Открыть Playground (“песочница” для языка программирования Swift, где мы можем поэксперементировать с ним);
* Создать новый проект Xcode
* Просмотреть созданные проекты (начать работу с созданным ранее проектом)



Рис.3. Окно запуска программы

Для того, чтобы это окно больше не открывалось при запуске, можно убрать галочку с параметра “Show this window when Xcode launches”.

*1.3 СОЗДАНИЕ ПРОЕКТА В СРЕДЕ ПРОГРАММИРОВАНИЯ XCODE*

Создаем новый проект, нажав на “Create a new Xcode project”.(рис.4)

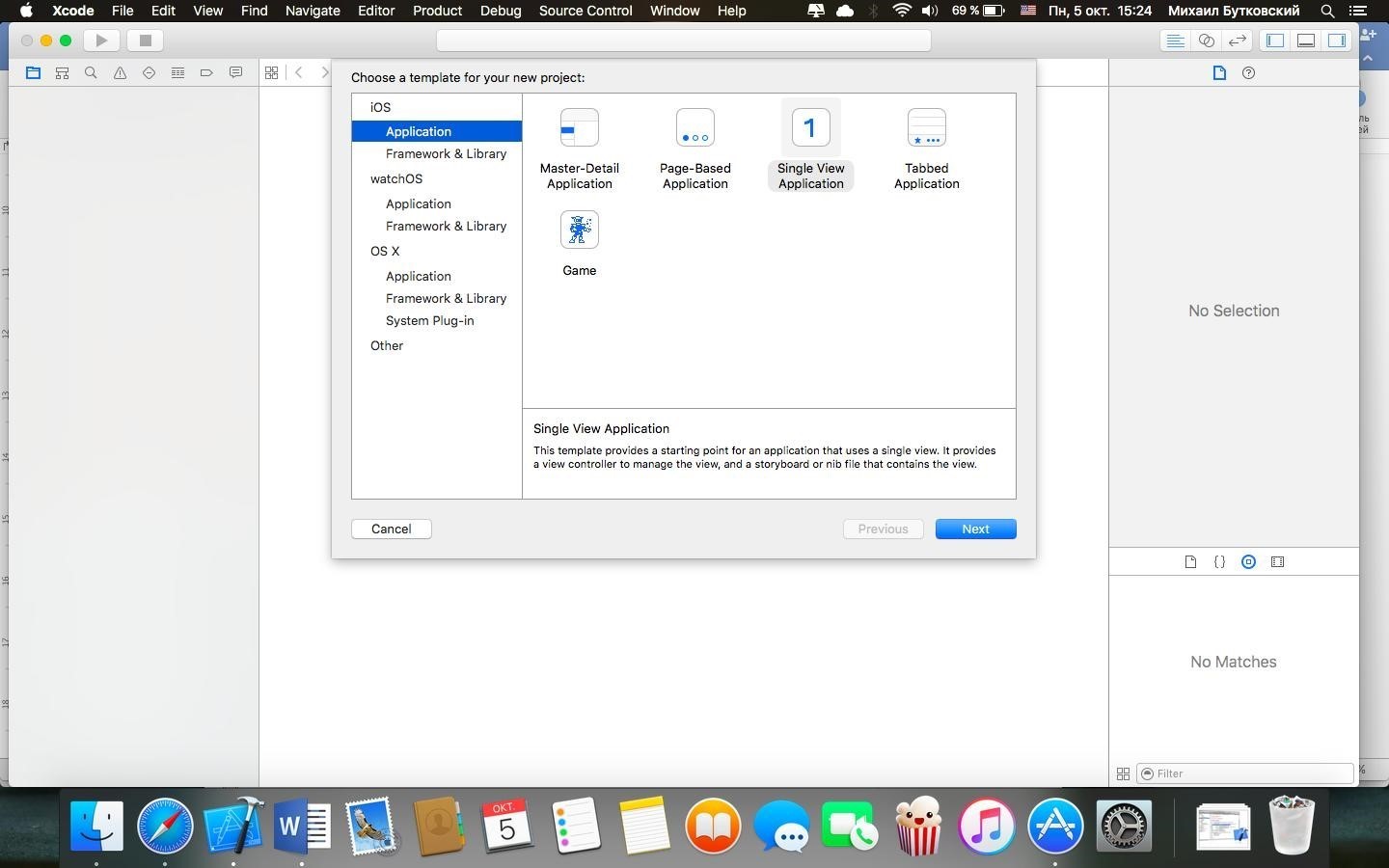


Рис.4. Создание нового проекта

Здесь мы можем выбрать шаблон, по которому будем создавать приложение для iOS (iPhone, iPad, iPod touch), watchOS (смарт-часы Apple Watch) или OS X (MacBook, MacBook Air, MacBook Pro, Mac mini, Mac Pro, iMac). Для каждой операционной системы имеется несколько шаблонов.

Выбираем iOS – Application – Single View Application (приложение с одним контроллером View Controller).

Нажимаем на кнопку “Next” и попадаем в окно выбора свойств нашего приложения (рис.5).

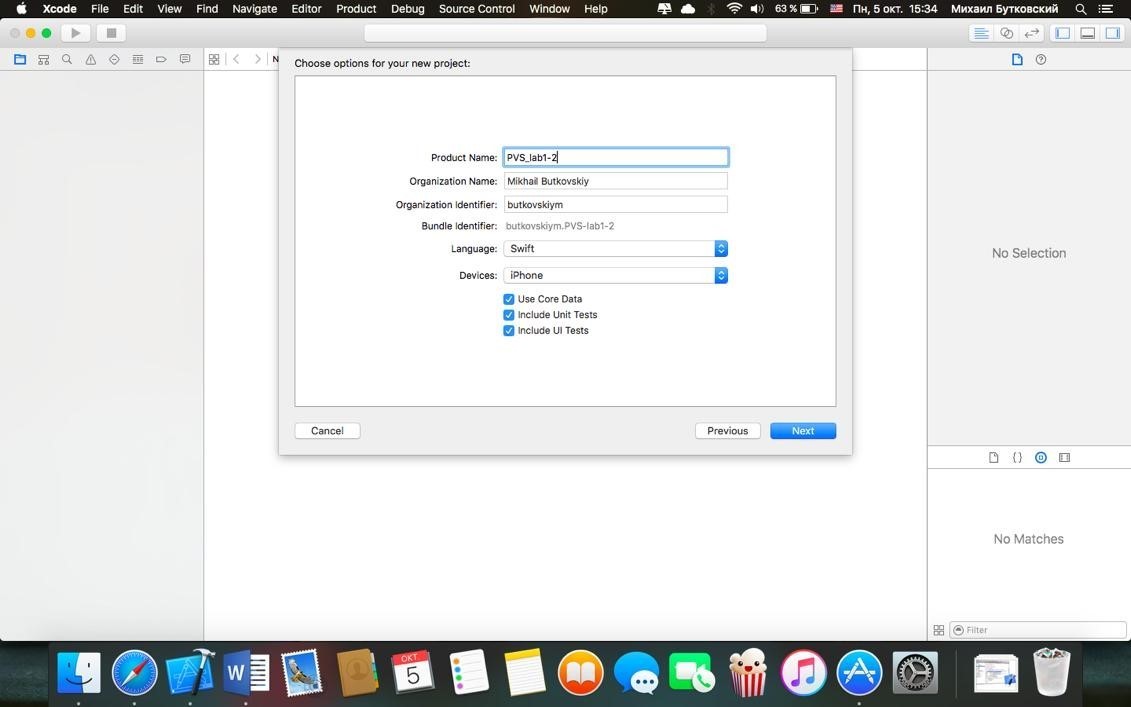


Рис.5. Окно выбора свойств

Здесь мы видим следующие параметры:

1. **Product Name** – название приложения
2. **Organization Name** – название компании разработчиков, либо имя индивидульного разработчика
3. **Organization Identifier** – идентификатор организации, он же используется как идентификатор сборки (Bundle Identifier)
4. **Language** – язык программирования
5. **Devices** – устройство, на которое будет нацелен проект (проект может быть нацелен сразу на iPhone и iPad – параметр Universal)
6. **Use Core Data** – использовать базу данных Core Data (встроенное в Xcode средство для работы с базами данных)
7. **Include Unit Tests** – включить модульное тестирование (процесс в программировании, позволяющий проверить на корректность отдельные модули исходного кода программы)
8. **Include UI Tests** – включить тестирование интерфейса пользователя

После того, как мы задали параметры для нашего приложения, нажимаем “Next”. Попадаем в окно выбора месторасположения нашего проекта, выбираем его и нажимаем “Create”.

*1.4 ОБЗОР ИНТЕРФЕЙСА СРЕДЫ ПРОГРАММИРОВАНИЯ XCODE И IOS SIMULATOR*

*Интерфейс среды программирования Xcode*

Все параметры приложения заданы, шаблон выбран – мы попадаем в сам интерфейс Xcode (рис.6).

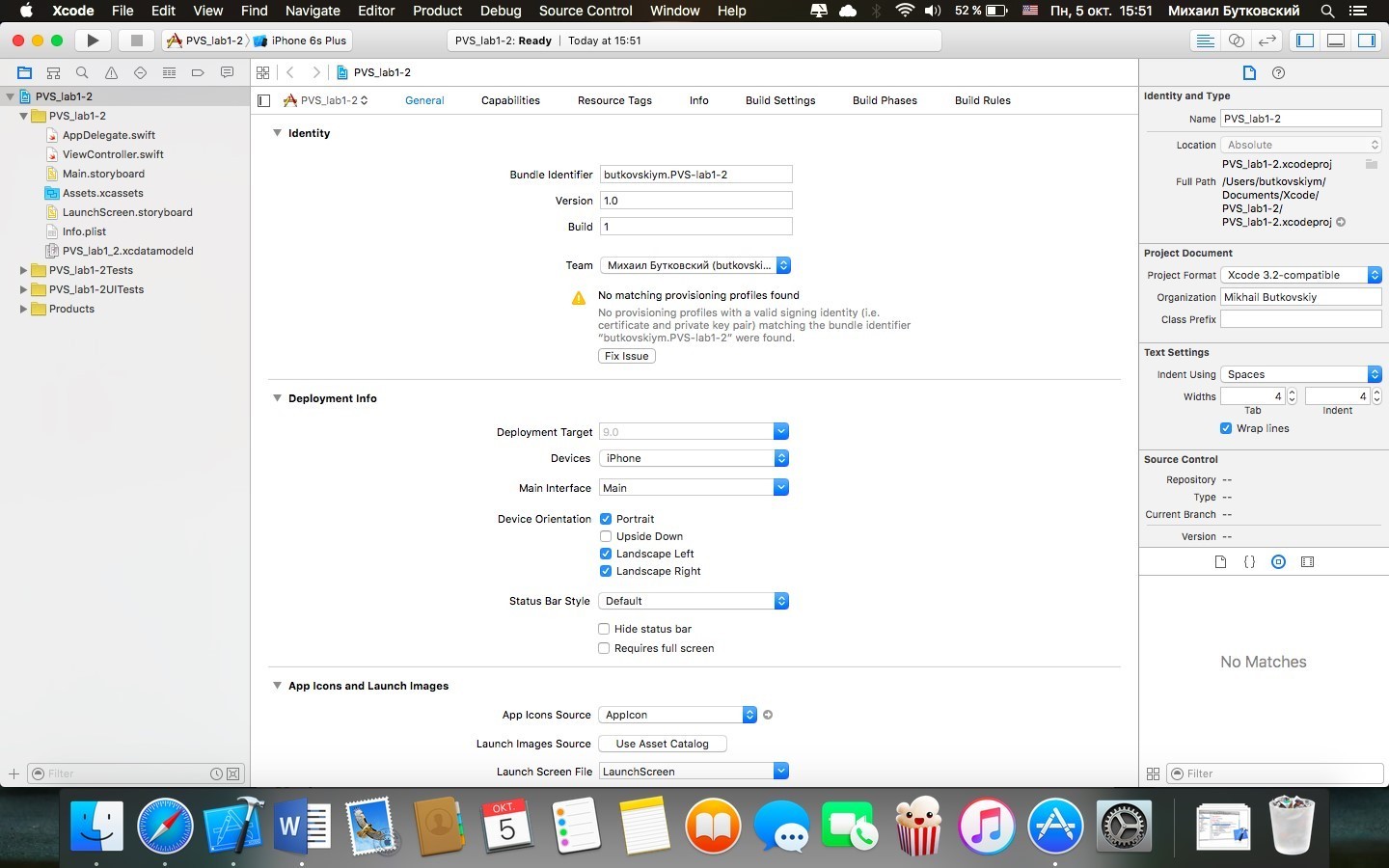


Рис. 6. Интерфейс Xcode

Главная часть интерфейса – это три области: (слева направо) ***область навигации***, ***область редактирования*** и ***область утилит***. В верхней части: (слева направо) кнопка компилирования и запуска приложения, выбор iOS симулятора и параметры схемы компилятора, строка состояния, кнопки редактора и кнопки вида.

*iOS Simulator*

iOS симулятор позволяет запускать написанные приложения на симуляторах различных устройств. Компания Apple настаивает на том, чтобы разработчики использовали инструменты только новейших операционных систем, в данном случае – iOS 9.

Для выбора устройства откройте список устройств в схеме компилятора. (рис.7). iOS Device в самой верхней части списка позволяет запустить приложение на вашем устройстве.

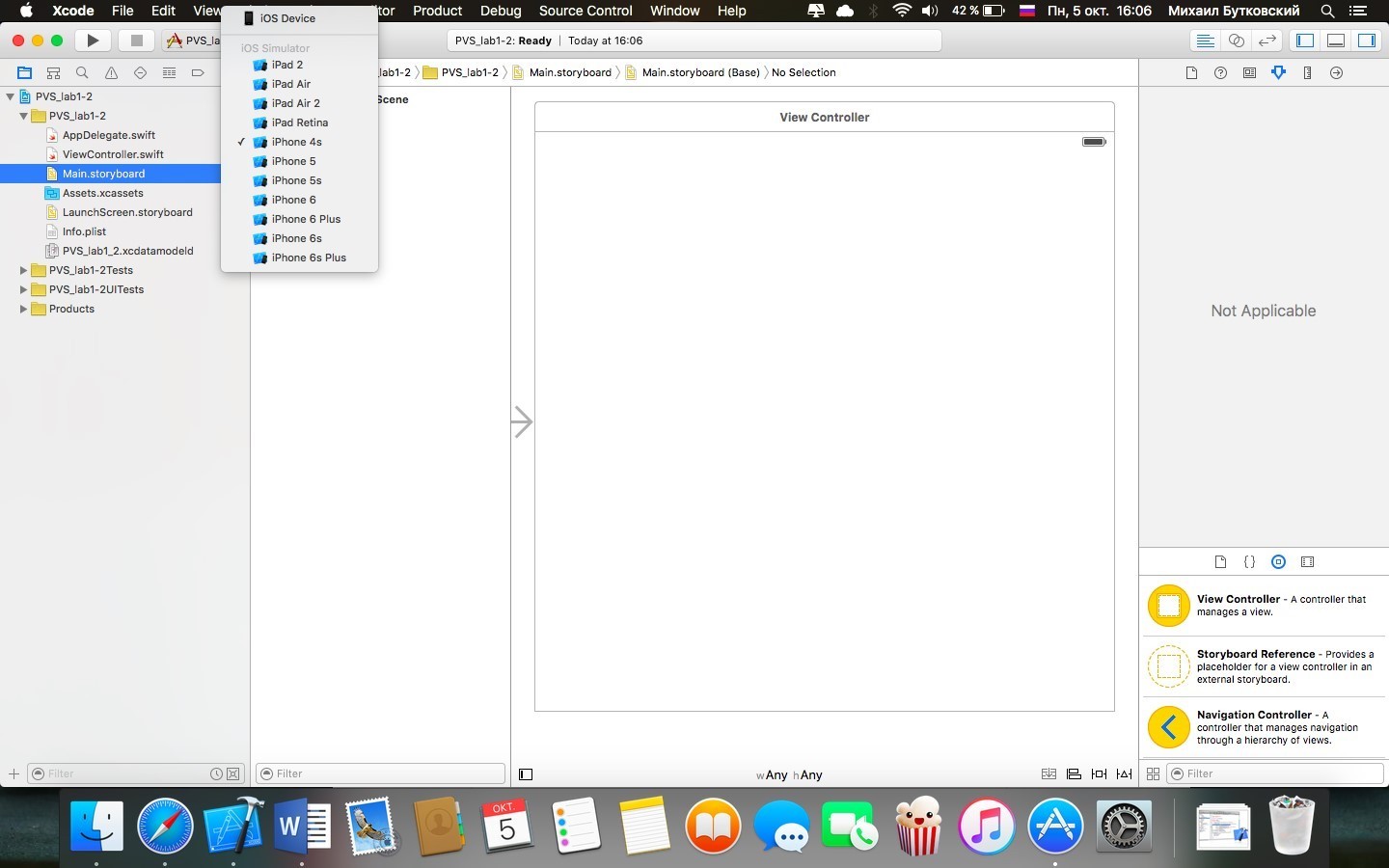


Рис. 7. Схема компилятора

*1.5 КОМПИЛЯЦИЯ И ЗАПУСК ПРИЛОЖЕНИЯ НА IOS SIMULATOR*

Для запуска приложения в симуляторе iOS-устройств выполните команду cmd+R на клавиатуре, либо нажмите на кнопку сборки и запуска проекта (Build and then run the current scheme) в правом верхнем углу интерфейса (рис.8).

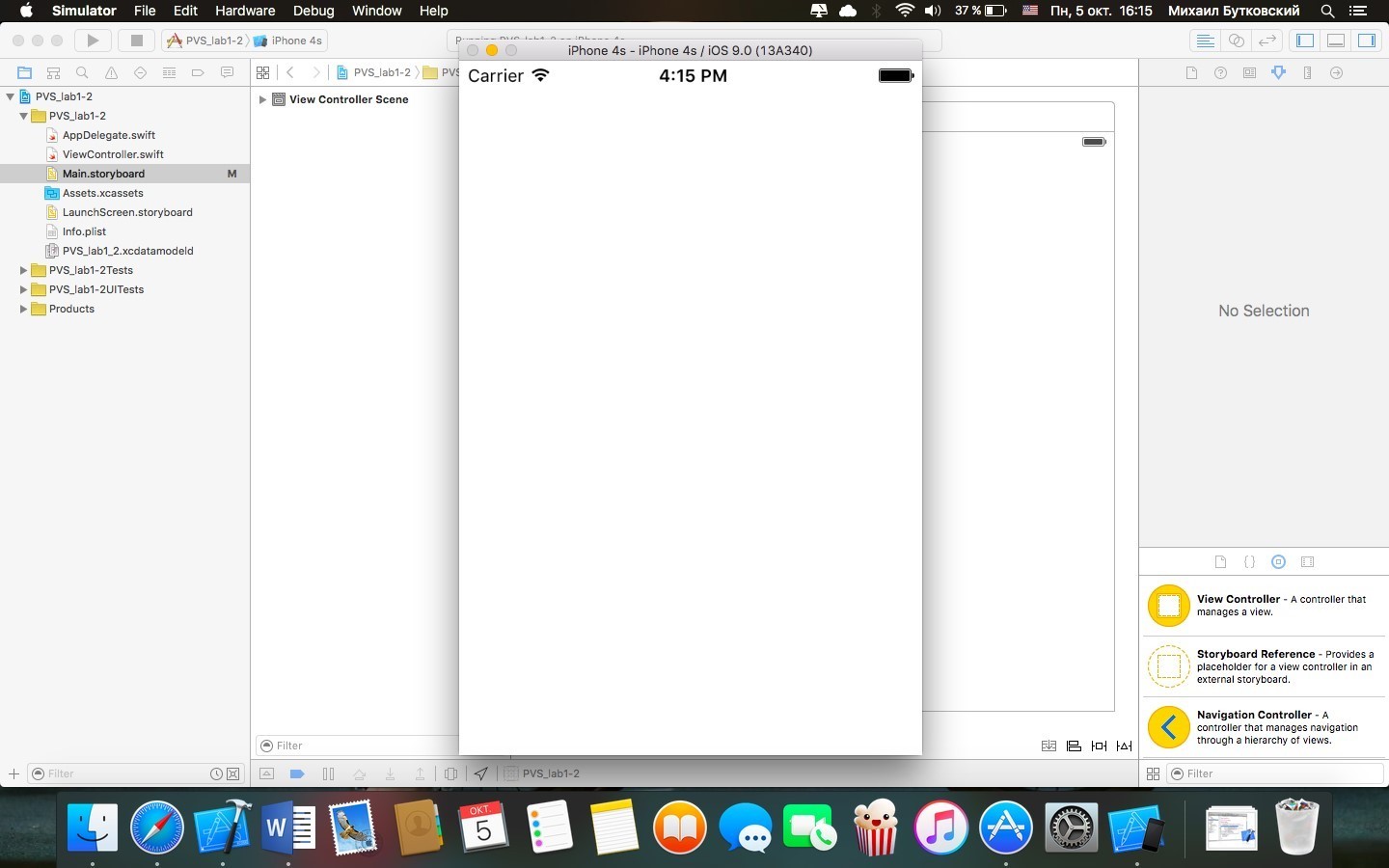


Рис. 8. Запуск приложения в симуляторе iOS

*1.6. ПРИВЯЗКА СВОЕГО APPLE ID К XCODE*

Для этого выбираем в верхней строке “Xcode” – “Preferences” – “Accounts”. В левом нижнем углу списка Apple IDs нажимаем “+” – “Add Apple ID…”, вводим данные своего Apple ID и нажимаем кнопку “Add”. Мы видим, как в списке Apple IDs появился наш аккаунтом (рис.10). Тем самым мы привязали наш Apple ID к Xcode и можем запускать приложения на любых iOS устройствах, которые привязаны к этому же Apple ID.

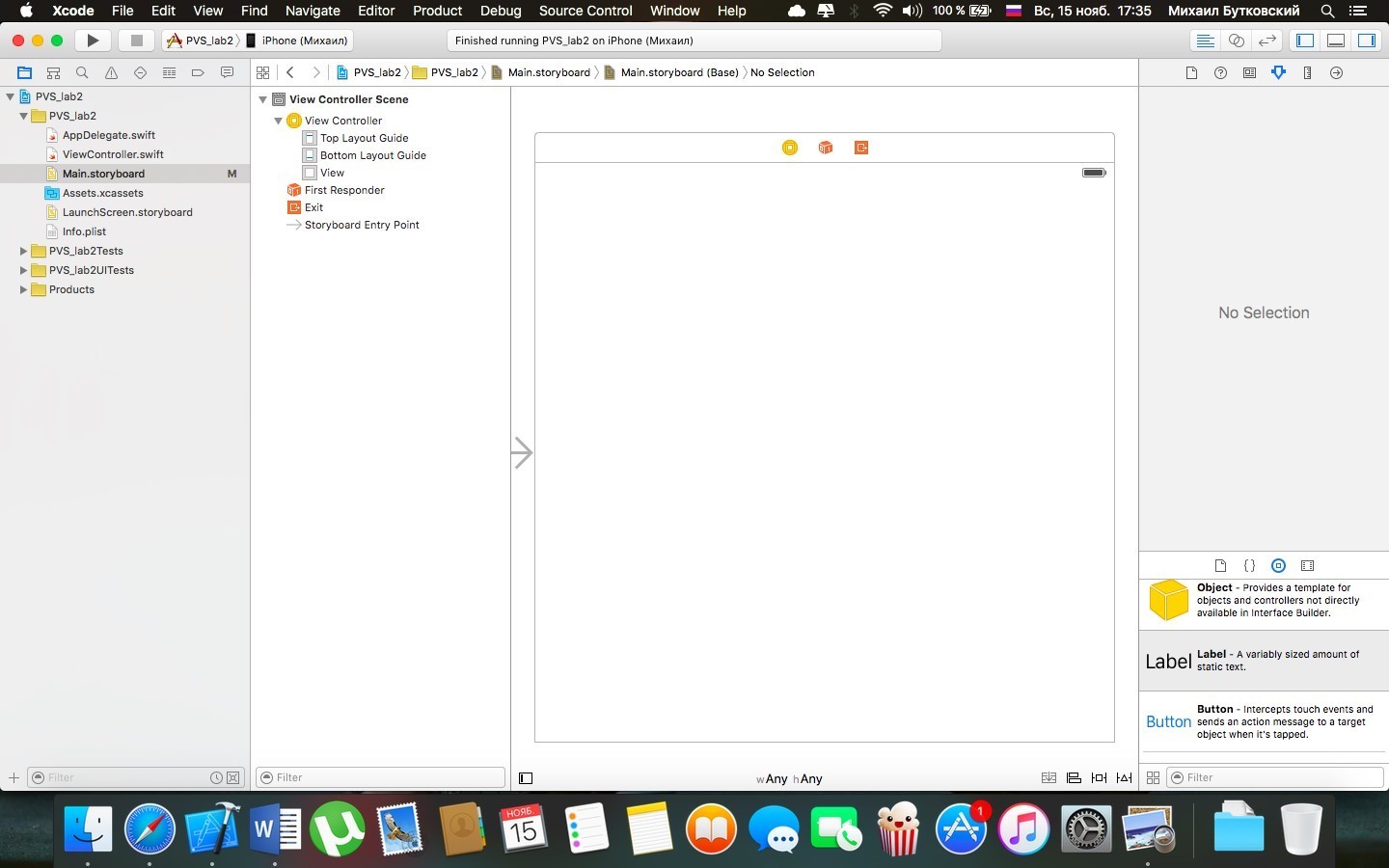


Рис. 9. Создание нового проекта

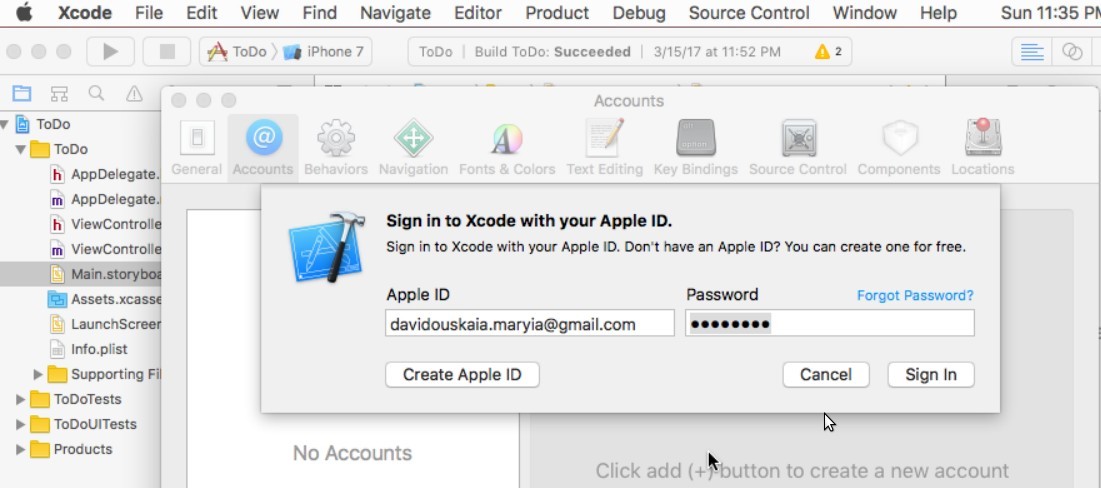


Рис. 10. Создание аккаунта

*ВОПРОСЫ*

1. Какие способы запуска симулятора iOS возможны?

Выбрать необходимое устройство в поле destination.

Запустить симулятор.

Так же симулятор представлен в виде отдельного приложения , которое можно найти у себя на Mac

1. Как управлять симулятором?

При запуске, симулятор полностью копирует поведение настоящего устройства. Т.е можно им пользоваться как обычным телефоном

## ЗАДАНИЕ 2. КОНСОЛЬНОЕ ПРИЛОЖЕНИЕ НА OBJECTIVE-C

**Цель** — познакомиться с синтаксисом языка Objective-C и разработать простейшее консольное приложение.

Изучить учебные материалы по созданию консольных приложений на языке Objective-C в Xcode:

1. [https://www.raywenderlich.com/2690-learn-to-code-ios-apps-1-welcome-to- programming](https://www.raywenderlich.com/2690-learn-to-code-ios-apps-1-welcome-to-programming) — **Изучить обязательно!!!**
2. [Подборка книг по Objective-C](https://edufpmi.bsu.by/mod/url/view.php?id=394)  — **Изучить обязательно!!!**
3. [https://iphonelearning.wordpress.com/2011/08/25/creating-first-console-app-using- xcode/](https://iphonelearning.wordpress.com/2011/08/25/creating-first-console-app-using-xcode/)
4. <https://andybargh.com/create-your-first-objective-c-application/>
5. <http://www.informit.com/articles/article.aspx?p=1315356>

**Реализовать приложение на Objective-C согласно варианту.**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №  п/п | Входные данные | Ожидаемые выходные данные | Действительные выходные данные | Тест пройден |
| 1 |  | 1 2 6 24 120 720 5040 40320 362880 3628800 | 1 2 6 24 120 720 5040 40320 362880 3628800 | Да |

**Опубликовать код приложения в git-репозиторий.**

*ВОПРОСЫ*

1. Какие расширения файлов у следуюших типов файлов?

C language source file - .c

C++ language source file - .cpp

Header file - .h

Objective-C source file - .m

Objective-C++ source file - .mm

Object (compiled) file - .o

1. Способы вывода на экран в Objective-C

NSLog,

printf

**ЗАДАНИЕ 3. КОНСОЛЬНЫЕ ПРИЛОЖЕНИЯ НА SWIFT:**

**СПОСОБЫ СОЗДАНИЯ**

**Цель** — познакомиться с консолью Swift REPL, компиляцией консольных приложений с помощью swiftc, созданием консольных приложений в xcode в playground; продемонстрируйте навыки работы с консолью Swift REPL и Playground, компиляцию приложений с помощью swiftc, создание и тестирование приложений в playground и создание консольных приложений. 1. Изучите примеры

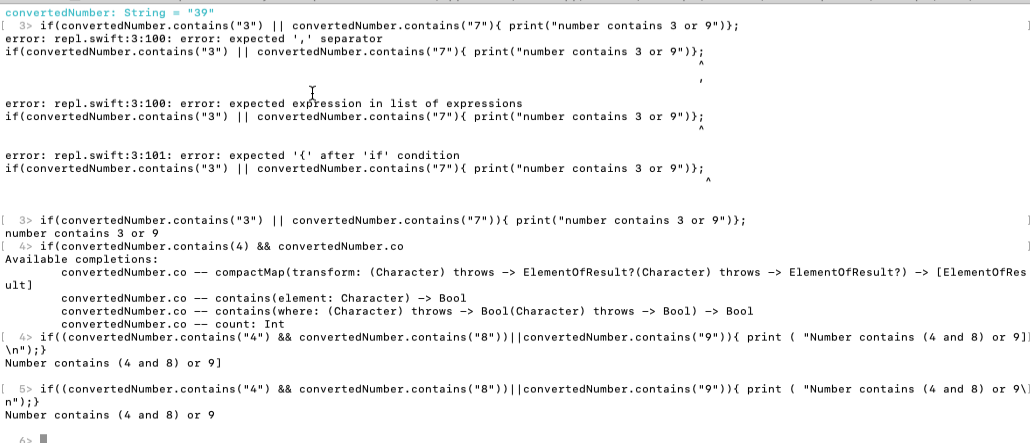
◦ из главы 2 книги Learn Swift <http://www.aidanf.net/learn-swift/running_code>.

◦<https://developer.apple.com/swift/blog/?id=18>

◦<https://swift.org/blog/swiftpm-repl-support/>

1. Разработать приложение на языке Swift согласно варианту в консоли Swift REPL и в Xcode Playground.
2. Продемонстрировать навыки работы с консолью Swift REPL, компиляцию приложений с помощью swiftc, создание и тестирование консольных приложений в playground.
3. Опубликовать код приложения в git-репозиторий.
4. Включить скриншоты и код приложений в отчет.

Фрагмент кода в REPL (swift):



Работа в playground :



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №  п/п | Входные данные | Ожидаемые выходные данные | Действительные выходные данные | Тест пройден |
| 1 | 43 | Содержит 3 или 7 | Содержит 3 или 7 | Да |
| 2 | 21 |  |  | Да |
| 3 | 84 | Содержит (4 и 8) или 9 | Содержит (4 и 8) или 9 | Да |

*ВОПРОСЫ*

## 1. Что такое swift REPL и в каких случаях рекомендуется использовать?

## Он позволяет нам сделать разбор написанного кода в режиме runtime. Удобно экспериментировать с небольшим по размеру кодом и выполнять debug.

## 2. Что такое playground? Какие задачи (проекты) можно создавать в

## playground?

## Playground, или в переводе на русский «песочница», представляет из себя отличное окружение для того, чтобы изучить Swift. Выглядит это так: вы вводите строку кода и моментально видите результат его выполнения. После того, как вы убедитесь, что написанный вами код дает нужный результат, вы можете просто взять и перенести его в свой проект.

## Например, при помощи Playground вы можете решить следующие задачи:

## освоить синтаксис Swift

## поэкспериментировать с новыми API, чтобы улучшить навыки программирования

## разработать новый алгоритм и понаблюдать за каждым шагом его выполнения

## провести простые арифметические вычисления.

## 3. Как компилировать приложение на языке swift в консоли bash?

## % swiftc -o file file\_name.swift

## % ./file\_name

## ЗАДАНИЕ 4. ПОДКЛЮЧЕНИЕ РЕПОЗИТОРИЯ И ОРГАНИЗАЦИЯ РАБОТЫ С ВЕТКАМИ В XCODE

Цель — изучить как работать с git-репозиторием в Xcode и подключить репозиторий для публикации кода в задании 4. Изучить примеры из статей:

1. <https://gist.github.com/quoha/4587902>
2. <https://www.raywenderlich.com/675-how-to-use-git-source-control-with-xcode-9>
3. [https://medium.com/xcblog/the-marriage-of-github-and-xcode-9-at-wwdc- 821b78eb6797](https://medium.com/xcblog/the-marriage-of-github-and-xcode-9-at-wwdc-821b78eb6797)
4. <http://blog.chrishannah.me/putting-your-xcode-project-on-github-the-easy-way/>
5. [https://medium.com/technology-nineleaps/integration-of-xcode-9-with-github- a350bf563278](https://medium.com/technology-nineleaps/integration-of-xcode-9-with-github-a350bf563278)

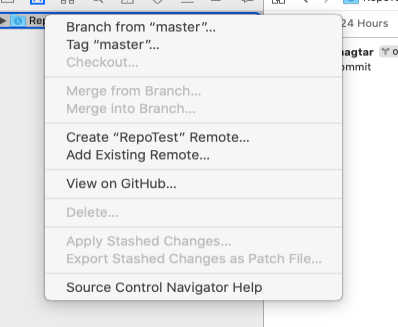
*ВОПРОСЫ*

* 1. Как создать локальный git-репозиторий для проектов в Xcode?

При запуске проекта выбрать опцию “ создать репозиторий”

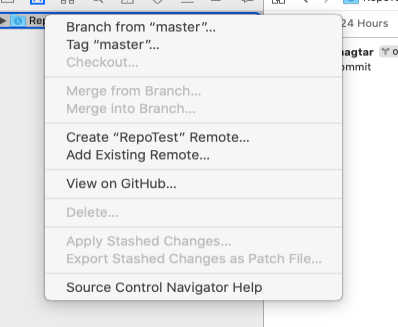
* 1. Как добавить внешний репозиторий в Xcode? Назвать два способа подключения.

1. Зайти в source control , выбрать clone.
   1. Как создать ветку репозитория в Xcode?



Выбрать Branch from master

* 1. Как слить ветки в Xcode?



Выбрать merge into Branch

## ЗАДАНИЕ 5. КОНСОЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ ОБРАБОТКИ ТЕКСТОВЫХ ДАННЫХ

Цель — разработать консольное приложение для обработки текстовых данных.

**Входные данные**

На вход программе подается список слов, разделенных пробелами.

**Выходные данные**

Согласно варианту задания.

Требуется реализовать приложение, которое будет преобразовывать строку, переданную в метод, использовать при реализации массив и словарь. Для вывода можно использовать функцию print(), это аналог функции printf().



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| №  п/п | Входные данные | Ожидаемые выходные данные | Действительные выходные данные | Тест пройден |
| 1 | Dsda Иван иван | иван | иван | Да |
| 2 | Ivan ivan galya  галя | галя | галя | Да |
| 3 | Река Лошадь Стол |  |  | Да |

Для решения задачи:

1. [Подборка книг по разработке приложений на языке Swift](https://edufpmi.bsu.by/mod/url/view.php?id=395)
2. Изучите примеры консольных приложений из статей:

◦ [https://www.raywenderlich.com/163134/command-line-programs-macos-tutorial- 2](https://www.raywenderlich.com/163134/command-line-programs-macos-tutorial-2)

◦<https://o7planning.org/en/10573/swift-tutorial-for-beginners>

1. Изучите типы коллекций согласно документации [https://developer.apple.com/library/content/documentation/Swift/Conceptual/ Swift\_Programming\_Language/CollectionTypes.html](https://developer.apple.com/library/content/documentation/Swift/Conceptual/Swift_Programming_Language/CollectionTypes.html) <https://developer.apple.com/documentation/swift/dictionary> и статьям

[https://iosrevisited.blogspot.com.by/2017/08/dictionariesios-in-swift-4-how-touse.html](https://iosrevisited.blogspot.com.by/2017/08/dictionariesios-in-swift-4-how-to-use.html)

<https://swift.org/blog/dictionary-and-set-improvements/>

<https://www.agnosticdev.com/content/how-work-dictionaries-swift-4>

1. Изучите методы сортировки строк [https://developer.apple.com/library/content/documentation/Cocoa/Conceptual/ Strings/Articles/SearchingStrings.html](https://developer.apple.com/library/content/documentation/Cocoa/Conceptual/Strings/Articles/SearchingStrings.html)
2. Создайте консольный проект в Xcode согласно варианту.
3. Опубликуйте код приложения в репозитории.

*ВОПРОСЫ*

1. В чем отличия языка программирования Swift от C и С++.

Отсутствуют указатели (то есть язык является более безопасным).

2. Какие способы отладки программ поддерживает среда xCode?

Breakpoint, Breakpoint - это точка, которая позволяет остановить программу в некоторый момент времени и выполнить некоторые действия над запущенной программой.

LLDB - консольный отладчик

3. Какие операторы языка swift Вы знаете?

Унарные, бинарные, тернарные (условный оператор: выражение ? действие1 : действие2).

Операторы сравнения, арифметические операторы, логические операторы.

4. Какие типы коллекций применяются в языке Swift и в чем их особенности?

Swift обеспечивает три основных типа коллекций - это Массивы, Множества и Словари для хранения коллекций значений.

5. Какие типы проектов xCode Вы знаете?

Single View Application – шаблон с одним готовым окном

Empty Application – пустой шаблон для создания приложения с

нуля

Tabbled Application – для проекта с несколькими вкладками

OpenGL Game – для игр на основе OpenGL.