

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой
Математического обеспечения ЭВМ

Абрамов Г.В.

29.05.2023 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.О.29 Архитектура мобильных устройств

1. Код и наименование направления подготовки/специальности:

02.03.02 Фундаментальная информатика и информационные технологии

2. Профиль подготовки/специализация: Инженерия программного обеспечения

3. Квалификация выпускника: бакалавр

4. Форма обучения: очная

5. Кафедра, отвечающая за реализацию дисциплины: МО ЭВМ

6. Составители программы: Болотова Светлана Юрьевна,

кандидат физико-математических наук, доцент

7. Рекомендована: НМС факультета ПММ, протокол № 7 от 26.05.2023 г.

8. Учебный год: _ 2025/2026

Семестр(ы): 5

9. Цели и задачи учебной дисциплины

Цели дисциплины:

- формирование знаний об основах архитектуры мобильных устройств и их компонентов;
- получение практических навыков разработки алгоритмических и программных решений в области программирования для мобильных устройств.

Задачи дисциплины:

- формирование и развитие навыков владения методами теории алгоритмов, методами системного и прикладного программирования;
- Формирование и развитие практического опыта разработки программного обеспечения для информационных систем;
- ознакомление с видами мобильных устройств;
- знакомство с архитектурой мобильных устройств и их компонентов;
- ознакомление с основными мобильными операционными системами;
- ознакомление с различными инструментами разработки программного обеспечения для мобильных устройств;
- ознакомление с основными методами и концепциями в области программирования мобильных приложений;
- знакомство с технологиями тестирования мобильных приложений;
- получение практических навыков по разработке полноценного мобильного приложения с применением всех изученных принципов, методик, методов и средств разработки мобильных приложений.

10. Место учебной дисциплины в структуре ООП: Дисциплина относится к обязательной части Блока1. Изучение данного курса должно базироваться на знаниях студентами объектно-ориентированного программирования.

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями) и индикаторами их достижения:

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
ОПК-3	Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям	ОПК-3.1.	Владеет методами теории алгоритмов, методами системного и прикладного программирования, основными положениями и концепциями в области математических, информационных и имитационных моделей.	Знать: методы теории алгоритмов, системного и прикладного программирования Уметь: определять и создавать информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем, разрабатывать программное обеспечение для информационных систем
		ОПК-3.2.	На основе соотнесения знаний в области программирования, интерпретации прочитанного, определяет и создает информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем.	Владеть: основными положениями и концепциями в области математических, информационных и имитационных моделей
		ОПК-3.3.	Имеет практический опыт разработки программного обеспечения для информационных систем.	

12. Объем дисциплины в зачетных единицах/час. — 3/108.

Форма промежуточной аттестации экзамен

13. Трудоемкость по видам учебной работы

Вид учебной работы		Трудоемкость		
		Всего	По семестрам	
			5	
Аудиторные занятия		48	48	
в том числе:	лекции	32	32	
	практические			
	лабораторные	16	16	
Самостоятельная работа		24	24	
в том числе: курсовая работа (проект)				
Форма промежуточной аттестации (экзамен – __ час.)		36	36	
Итого:		108	108	

13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	н р з е с ц Содержание а д л а д и и п л и н ы	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК *
1. Лекции			
1.1	Введение в разработку мобильных приложений	Архитектура платформы iOS, компоненты платформы. Знакомство с XCode.	https://edu.vsu.ru/course/view.php?id=5157
1.2	Шаблон проектирования MVC.	Шаблон проектирования MVC. Классы, структуры.	https://edu.vsu.ru/course/view.php?id=5157
1.3	Autolayout. Контроль доступа. Перечисления.	CountableRange чисел с плавающей точкой, кортежи, вычисляемые свойства, управление доступом, assertions, enum .	https://edu.vsu.ru/course/view.php?id=5157
1.4	Протоколы. Oprionals. Расширения	Протоколы. Oprionals. Расширения.	https://edu.vsu.ru/course/view.php?id=5157
1.5	Строки. Функции. Замыкания.	Строки с атрибутами. Функции как типы. Замыкания. Обработка ошибок в Swift. Использование Any в Swift. «Кастинг типа» с помощью оператора as? NSObject, NSNumber, Date, Data.	https://edu.vsu.ru/course/view.php?id=5157
1.6	Views	Views. Создание пользовательского subclass UIView.	https://edu.vsu.ru/course/view.php?id=5157
1.7	Множественные MVC	TabBarController, SplitViewController и NavigationController.	https://edu.vsu.ru/course/view.php?id=5157
1.8	Обработка жестов	Жесты pinch, pan, rotation, swipe, tap. Обработчики жестов.	https://edu.vsu.ru/course/view.php?id=5157
3. Лабораторные занятия			
3.1	Шаблон проектирования MVC. Множественные MVC.	Демонстрационный пример. Создание проекта Project в Xcode 9 Построение пользовательского интерфейса (UI) iOS симуляторы print (вывод на консоль, используя \ () нотацию) Определение класса в Swift, включая определение переменных экземпляра класса и методов Связывание свойств (переменных экземпляра	https://edu.vsu.ru/course/view.php?id=5157

		<p>класса) в Swift коде с элементами пользовательского интерфейса UI (Outlets) Привязка элементов UI к методам в коде Swift (Actions) Доступ к iOS документации из кода Автоматическое выполнение кода при каждом изменении значения свойства Optionals (? , неявное разворачивание путем декларирования со знаком !, явное разворачивание с помощью ! и if let) Array Создание модели игры Concentration.</p>	
3.2	Рисование в iOS.	<p>Рисование с помощью Core Graphics и UIBezierPath. Режим contentMode у UIView. Рисование с прозрачностью. Больше ключей для словаря строки с атрибутами NSAttributedString ... UIFont и NSMutableParagraphStyle UIFontMetrics масштабирование шрифтов определенного стиля согласное с пользовательскими настройками (Setting) в Larger Text Управление subviews в вашем пользовательском UIView Использование.isHidden Аффинные преобразования CGAffineTransform UIView Приоритеты ограничений (Constraint Priority) в системе Autolayout Assets.xcassets и рисование с помощью UIImage @IBDesignable и @IBInspectable Отображение изображений UIImage в Interface Builder</p>	https://edu.vsu.ru/course/view.php?id=5157

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (количество часов)				
		Лекции	Практические	Лабораторные	Самостоятельная работа	Всего
1	Введение в разработку мобильных приложений	2				2
2	Шаблон проектирования MVC.	4		2		6
3	Autolayout. Контроль доступа. Перечисления.	4			2	6
4	Протоколы. Optionals. Расширения	4			6	10
5	Строки. Функции. Замыкания.	4			6	10
6	Views	2				2
7	Множественные MVC	8		6	6	20
8	Обработка жестов	2				2
9	Рисование в iOS	2		8	4	14
	Итого:	32		16	24	72

14. Методические указания для обучающихся по освоению дисциплины

рекомендации обучающимся по освоению дисциплины: работа с конспектами лекций, презентационным материалом

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины (список литературы оформляется в соответствии с требованиями ГОСТ и используется общая сквозная нумерация для всех видов источников)

а) основная литература:

№ п/п	Источник
1	Соколова, В. В. Разработка мобильных приложений : учебное пособие / В. В. Соколова. — Томск : ТПУ, 2014. — 176 с. — ISBN 978-5-4387-0369-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/82830 (дата обращения: 27.03.2021). — Режим доступа: для авториз. пользователей.
2	Грэхем, Л. Разработка через тестирование для iOS / Л. Грэхем ; перевод с английского А. Н. Киселев. — Москва : ДМК Пресс, 2013. — 272 с. — ISBN 978-5-94074-863-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/63183 (дата обращения: 27.03.2021). — Режим доступа: для авториз. пользователей.

б) дополнительная литература:

№ п/п	Источник
1	Райфельд, М. А. Системы и сети мобильной связи : учебное пособие / М. А. Райфельд, А. А. Спектор. — Новосибирск : НГТУ, 2019. — 96 с. — ISBN 978-5-7782-3833-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/152245 (дата обращения: 27.03.2021). — Режим доступа: для авториз. пользователей.
2	Программируем для iPhone и iPad / Д. Пайлон, Т. Пайлон. - Питер, 2012. ISBN 978-5- 459-00375-8.
3	Разработка и продажа программ для iPhone и iPad / Дмитрий Елисеев. - БХВ- Петербург, 2012. ISBN 978-5-9775-0687-8.

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет)*:

№ п/п	Ресурс
1.	www.staford.edu Курс лекций Ipad and Iphone Application Development
2.	Open handset alliance, http://www.openhandsetalliance.com/ .
3.	

* Вначале указываются ЭБС, с которыми имеются договора у ВГУ, затем открытые электронно-образовательные ресурсы, онлайн-курсы, ЭУМК

16. Перечень учебно-методического обеспечения для самостоятельной работы (учебно-методические рекомендации, пособия, задачки, методические указания по выполнению практических (контрольных), курсовых работ и др.)

№ п/п	Источник
1	Mayer R. Professional Android 4 Application Development. – Willeyand Sons, 2012. <i>Соколова</i>
2	<i>В. В. Разработка мобильных приложений : учебное пособие. — Томск: Изд-во ТПУ, 2014. — 175 с.: ил.</i>
3	Таненбаум, Э. Современные операционные системы [Текст] = Modern Operating Systems / Э. Таненбаум ; пер. с англ. Н. Вильчинского, А. Лашкевич. - 3-е изд. - Санкт-Петербург : Питер, 2012. - 1115 с.

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ), электронное обучение (ЭО), смешанное обучение):

При реализации дисциплины используются модульно-рейтинговая и личностно-ориентированные технологии обучения (ориентированные на индивидуальность студента, компьютерные и коммуникационные технологии). В рамках дисциплины предусмотрены следующие виды лекций: информационная, лекция-визуализация, лекция с применением обратной связи.

Дисциплина реализуется с применением электронного обучения и дистанционных образовательных технологий, для организации самостоятельной работы обучающихся используется онлайн-курс, размещенный на платформе Электронного университета ВГУ (LMS moodle), а также другие Интернет-ресурсы, приведенные в п.15в.

18. Материально-техническое обеспечение дисциплины:

Лекционная аудитория должна быть оборудована учебной мебелью, компьютером, мультимедийным оборудованием (проектор, экран, средства звуковоспроизведения), допускается переносное оборудование.

Практические занятия должны проводиться в специализированной аудитории, оснащенной учебной мебелью и персональными компьютерами APPLE Mac Pro MD772RU/A Xeon W3565 с доступом в сеть Интернет (компьютерные классы, студии), мультимедийным оборудованием (мультимедийный проектор, экран, средства звуковоспроизведения), Число рабочих мест в аудитории должно быть таким, чтобы обеспечивалась индивидуальная работа студента на отдельном персональном компьютере.

Для самостоятельной работы необходимы компьютерные классы, помещения, оснащенные компьютерами с доступом к сети Интернет.

Программное обеспечение:

- Android studio (свободное и/или бесплатное ПО)
- LibreOffice (свободное и/или бесплатное ПО)
- Microsoft Visual Studio Community Edition (свободное и/или бесплатное ПО)
- Adobe Reader (свободное и/или бесплатное ПО)
- Xcode

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1.	Введение в разработку мобильных приложений	ОПК-3	ОПК-3.1 ОПК-3.2	Собеседование
2.	Шаблон проектирования MVC.	ОПК-3	ОПК 3.2 ОПК-3.3	Лабораторная работа
3.	Autolayout. Контроль доступа. Перечисления.	ОПК-3	ОПК-3.3	Лабораторная работа
4.	Протоколы. Optionals. Расширения	ОПК-3	ОПК-3.1 ОПК-3.3	Лабораторная работа
5.	Строки. Функции. Замыкания.	ОПК-3	ОПК-3.1 ОПК-3.2 ОПК-3.3	Лабораторная работа
6.	Views	ОПК-3	ОПК 3.1 ОПК-3.3	Лабораторная работа
7.	Множественные MVC		ОПК-3.3	Лабораторная работа
8.	Обработка жестов		ОПК-3.1 ОПК-3.2 ОПК-3.3	Лабораторная работа
9.	Рисование в iOS		ОПК-3.3	Лабораторная работа
Промежуточная аттестация форма контроля - экзамен				Перечень вопросов

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1 Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

Лабораторная работа.

Перечень лабораторных работ

Заставьте работать игру Концентрация, демонстрируемую на лекции. Печатайте весь код, не пользуйтесь копированием и вставкой кода откуда-то. Добавьте больше карт в вашу игру. Добавьте на ваш UI кнопку “New Game”, которая заканчивает текущую игру и начинает новую. В данный момент карты в Моделе не рандомизированы (именно поэтому в вашем UI парные карты всегда лежат на тех же самых местах). Перетасуйте карты в методе `init()` класса `Concentration`. Введите в игру концепцию “Тема” (“`theme`”). Тема `theme` определяет множество эмоджи, из которого выбираются эмоджи для карт. Все эмоджи в определенной теме должны иметь отношение к этой теме. Ваша игра должна, по крайней мере, иметь 6 различных тем, и темы должны выбираться случайно каждый раз при старте новой игры. Ваша архитектура должна давать возможность добавлять новую тему одной строкой кода. Добавьте на ваш UI метку для счета в игре (`score label`). Счет в игре формируется добавлением 2-х очков за каждое совпадение и штрафом в 1 очко за каждое несовпадение ранее увиденной карты. Отслеживание числа переворотов карт `flipCount` определено НЕ принадлежит вашему `Controller` в правильной MVC архитектуре. Исправьте это. Весь новый добавленный UI должен правильно располагаться и выглядеть хорошо в портретном режиме на iPhone X.

Реализуйте игру Set в версии соло (для одного игрока). Разместите на экране по крайней мере 24 карты игры Set. В Set все карты всегда лежат “лицом” вверх. При старте сдайте только 12 карт. Они могут появиться где угодно на экране (то есть обязательно их выравнивать по верху или по низу экрана или как-то еще; при старте они могут быть рассеяны, если хотите), но они не должны перекрываться. Вам необходимо также иметь кнопку “Deal 3 More Cards” (Сдай еще 3 карты) (согласно правилам игры Set). Разрешите пользователю выбирать карты касанием для того, чтобы попытаться составить Set. На ваше усмотрение, как показывать “выбор” в вашем UI. Некоторые идеи того, как это можно сделать представлены ниже в подсказках. Также обеспечьте возможность переход из состояние “выбрано” (`selected`) в состояние “не выбрано” (`deselected`) (но только когда 1 или 2 (не 3) карты выбраны в данный момент). После того, как выбраны 3 карты, вы должны дать пользователю индикацию, совпали ли эти 3 карты или нет (согласно правилам игры Set). Вы можете сделать это с помощью цвета или как хотите, но пользователю должно быть понятно, совпали эти 3 карты или нет... Когда выбрана новая карта и есть уже 3 выбранных (`selected`) и не совпавших Set карты, сделайте эти 3 не совпавших карты не выбранными (`deselected`), а новую карту выбранной (`selected`). Согласно правилам игры Set, когда выбрана новая карта и есть уже 3 совпавших (`matching`) и выбранных (`selected`) Set карты, замените эти 3 совпавших (`matching`) Set карты новыми из колоды в 81 Set карту (опять, смотрите правила игры Set и что собой представляет колода Set карт). Если колода пуста, то совпавшие (`matching`) Set карты не могут быть заменены, но они могут быть скрыты (`hidden`) в вашем UI. Если вновь выбранная карта является одной из 3-х совпавших (`matching`) Set карт, то никакие карты не должны быть выбранными (`selected`) (так как вновь выбранная карта либо будет заменена, либо будет больше невидима на UI). Когда кнопка “Deal 3 More Cards” (Сдай еще 3 карты) нажата, то либо а) происходит замена выбранных карт, если они совпали, либо б) добавляются 3 карты в игру. Кнопка “Deal 3 More Cards” (Сдай еще 3 карты) должна быть недоступна, если а) больше нет карт в Set колоде или б) больше нет места на UI, чтобы принять еще 3 карты (заметьте, что всегда есть место для размещения еще 3-х карт, если выбранные в данный момент карты совпали (`match`), так как они заменяются). Вместо рисования Set карт в классической форме, мы будем использовать эти 3 символ ▲ ● ■ и использовать атрибуты в `NSAttributedString` для соответствующего их рисования (то есть цвета и затенение (`shading`)). И таким образом, ваши карты могут быть просто кнопками `UIButton`s. Используйте метод, который берет в качестве аргумента замыкание как значимую часть вашего решения. Используйте перечисление `enum` как значимую часть вашего решения. Добавьте осмысленное расширение `extension` к некоторым структурам данных как значимую часть вашего решения. Ваш UI должен иметь прекрасно расположенные UI элементы и хорошо выглядеть (по крайней мере в портретном режиме, желательно также и в ландшафтном режиме, хотя это не обязательно) на любом iPhone 7 или старше. Это означает, что вам следует использовать несколько простых приемов работы с `Autolayout`, включая `Stack Views`.

Описание технологии проведения

Каждая лабораторная работа выполняется на основе задания и соответствующей лекции. После выполнения задания на лабораторную работу каждый студент должен выполнить те же действия, но уже по своей теме, которая относится к домашнему заданию по дисциплине. Таким образом, после каждой лабораторной работы формируются необходимые части/знания для выполнения домашнего задания.

Требования к выполнению заданий (или шкалы и критерии оценивания)
Каждая лабораторная работа оценивается по принципу «зачет/незачет»

«Зачет» ставится, если сделано верно не менее 80% задания

«Незачет» ставится, если сделано верно менее 80% задания

20.2 Промежуточная аттестация

Промежуточная аттестация по дисциплине осуществляется с помощью следующих оценочных средств:

Собеседование по экзаменационным билетам (по билетам к зачету)

Примеры вопросов к экзамену

1. Объясните идею паттерна MVC на примере простейшего калькулятора.
2. Приведите пример создания кортежа с 2 элементами (строкой и целым числом) со значениями “hello” и 12.
3. Как называется такое свойство?

```
var prop: Int {  
    get { ... }  
    set (newValue) {...}  
}
```

4. Какой тип имеет переменная a?

```
let a: Int?
```

5. Напишите код, который распечатает значение этой переменной.

6. Исправьте ошибку в данном коде.

```
enum Item {  
    case A (number: Int)  
    case B (part: Double)  
    case C (description: String)  
    case D
```

```
func switchToBeingCookie () {  
    self = .D  
}
```

7. В чем заключается ошибка в данном коде?

```
protocol A {  
    mutation func f1()  
}
```

```
class B: A {  
    func f1()  
    func f2()  
}
```

```
let x: B = B()  
var y: A = x  
y.f2()
```

8. Что означает lazy в данном коде?

```
lazy var a = A()
```

8. Что такое циклическая ссылка в памяти? Использование какого типа ссылок (string, weak, unowned) позволяет цикл разорвать?

9. Что происходит в этом коде?

```
if let cvc = vc as? ConcentrationViewController {...}
```

10. На экране находится прямоугольник, который вы повернули. С помощью какого/каких методов можно заставить прямоугольник перерисоваться:

```
func draw(_ rect: CGRect);  
func setNeedsDisplay(_ rect: CGRect);  
func redraw(_ rect: CGRect);  
func update();  
func setNeedsDisplay( )?
```

Описание технологии проведения

Экзамен проходит в письменной форме

Требования к выполнению заданий, шкалы и критерии оценивания

Критерии оценки:

оценка «отлично» выставляется обучающемуся, если правильный ответ дан не менее чем на 75% вопросов;

оценка «хорошо» выставляется обучающемуся, если правильный ответ дан не менее чем на 50% вопросов;

оценка «удовлетворительно» выставляется обучающемуся, если правильный ответ дан не менее чем на 30% вопросов;

оценка «неудовлетворительно» выставляется обучающемуся, если правильный ответ дан менее чем на 30% вопросов.