

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

УТВЕРЖДАЮ

Заведующий кафедрой
Математического обеспечения ЭВМ



Абрамов Г.В.
23.03.2024г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Б1.В.ДВ.03.04.04 Программирование для мобильных устройств

1. Код и наименование направления подготовки/специальности:

01.03.02 Прикладная математика и информатика

2. Профиль подготовки/специализация: Прикладная математика и компьютерные технологии

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 № 5 от 22.03.2024

8. Учебный год: 2027-2028 Семестр(ы): 7

9. Цели и задачи учебной дисциплины

Цель изучения дисциплины: является овладение современными понятиями и способами написания и сопровождения программ на языке Swift, необходимыми в профессиональной практической деятельности. Задачи учебной дисциплины: формирование способностей применять эффективные структуры данных при проектировании программных продуктов, а также - современные эффективные технологии разработки мобильных приложений, работающих под управлением платформа iOS, используя для этого современные средства разработки.

10. Место учебной дисциплины в структуре ООП: учебная дисциплина относится к формируемой участниками образовательных отношений части Блока 1.

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
ПК-12	Способен составлять формализованные описания решений, разрабатывать алгоритмы решения поставленных задач, проектировать структуры данных, базы данных и интерфейсы	ПК-12.1.	Владеет языками формализации функциональных спецификаций, методами и приемами формализации задач, методами и средствами проектирования информационных ресурсов, баз данных и интерфейсов, основами web-технологий, программными средствами и платформами для разработки web-ресурсов	<p>Знать: современные математические и компьютерные методы в задачах анализа динамических систем и оптимизации</p> <p>Уметь: использовать современные технологии для решения задач анализа динамических систем, оптимизации и оптимального управления</p> <p>Владеть: навыками работы с пакетами прикладных программ для решения задач анализа динамических систем, оптимизации и оптимального управления</p>
ПК-13	Способен разрабатывать и применять алгоритмические и программные решения в области системного и прикладного программного обеспечения	ПК-13.1	Демонстрирует знания методологии разработки программного обеспечения и технологии программирования, методологии теории проектирования и баз данных	<p>Знать: методов математического, функционального и системного анализа, применяемые для решения задач моделирования, исследования и синтеза систем автоматического управления техническими объектами</p> <p>Уметь: разрабатывать и адаптировать прикладное программное обеспечение для автоматизированных систем управления</p> <p>Владеть: навыками формулирования математических задач исследований, выбора, анализа, интерпретации и представления результатов исследований</p>
		ПК-13.2	Проектирует информационные ресурсы, тестирует их с точки зрения пользователя удобства, программирует приложения и создает программные прототипы решения прикладных задач	<p>Знать: приемы алгоритмизации поставленных задач, стандартные алгоритмы в соответствующих областях</p> <p>Уметь: пользоваться пакетами прикладных программ, нормативно-технической документацией по процессам управления</p> <p>Владеть: методами и приемами формализации и алгоритмизации поставленных задач, стандартными алгоритмами в соответствующих областях, пакетами прикладных программ, нормативно-технической документацией по процессам управления</p>

Форма промежуточной аттестации зачет

13. Трудоемкость по видам учебной работы

Вид учебной работы		Трудоемкость		
		Всего	По семестрам	
			7	
Аудиторные занятия		32	32	
в том числе:	лекции	16	16	
	практические			
	лабораторные	16	16	
Самостоятельная работа		40	40	
в том числе: курсовая работа (проект)				
Форма промежуточной аттестации (экзамен – __ час.)				
Итого:		72	72	

13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	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=18637
1.2	Шаблон проектирования MVC	Шаблон проектирования MVC. Классы, структуры.	https://edu.vsu.ru/course/view.php?id=18637
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=18637
1.4	Протоколы. Optional. Расширения	Протоколы. Optional. Расширения.	https://edu.vsu.ru/course/view.php?id=18637
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=18637
1.6	Views	Views. Создание пользовательского subclass UIView.	https://edu.vsu.ru/course/view.php?id=18637
3. Лабораторные занятия			
3.1	Шаблон проектирования 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=18637

		<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 развертывание путем декларирования со знаком ! , явное развертывание 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=18637

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	4
2	Шаблон проектирования MVC.	2		2	2	6
3	AutoLayout. Контроль доступа. Перечисления.	2		2	6	10
4	Протоколы. Optionals. Расширения	2		2	6	10
5	Строки. Функции. Замыкания.	2		4	6	12
6	Views	2		2	8	12
9	Рисование в iOS	4		4	10	18
	Итого:	16		16	40	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.
2	Соколова В. В. Разработка мобильных приложений : учебное пособие. — Томск: Изд-во ТПУ, 2014. — 175 с.: ил.
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:

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

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

Программное обеспечение: Xcode

Материально-техническое обеспечение:

Моноблок Apple iMac MD093RU/A (14 шт.): процессор Intel Core i5 (2.70 GHz), оперативная память 8 Гб, HDD 1 Тб, видеокарта GeForce GT640M 512Мб, диагональ экрана 21,5"

Компьютер APPLE Mac Pro MD772RU/A Xeon W3565 в составе:

системный блок APPLE: процессор Intel Xeon W3565, оперативная память 8Гб, HDD 2Тб, видеокарта AMD Radeon HD 5770

Коммутатор HP ProCurve Switch 1400-24G

Мультимедиа-проектор BENQ MH535

Доска магнитно-маркерная на стенде (100x150см), 2-сторонняя, BRAUBERG PREMIUM

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	ПК-12	ПК-12.1	Собеседование
2.	Шаблон проектирования MVC.	ПК-12	ПК-12.1	Лабораторная работа
3.	Autolayout. Контроль доступа. Перечисления.	ПК-13	ПК-13.2	Лабораторная работа
4.	Протоколы. Originals. Расширения	ПК-13	ПК-13.2	Лабораторная работа
5.	Строки. Функции. Замыкания.	ПК-13	ПК-13.2	Лабораторная работа
6.	Views	ПК-13	ПК-13.1 ПК-13.2	Лабораторная работа
9.	Рисование в iOS	ПК-12 ПК-13	ПК-12.1 ПК-13.1 ПК-13.2	Лабораторная работа
Промежуточная аттестация форма контроля - зачет				Контрольное тестирование

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

Реализуйте 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 theme 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 определенно 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. 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:

Контрольное тестирование

Примеры теста

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

Тестирование проходит в письменной форме

Требования к выполнению заданий, шкалы и критерии оценивания

Критерии оценки:

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

«не зачтено» выставляется обучающемуся, если правильный ответ дан менее чем на 50% вопросов.