



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение
высшего образования

«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ФГБОУ ВО «ИГУ»

Факультет бизнес-коммуникаций и информатики

Кафедра естественнонаучных дисциплин

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине Б1.В.14 Основы мобильной разработки

направление подготовки 09.03.03 Прикладная информатика

направленность (профиль) Прикладная информатика (разработка программного обеспечения)

Одобен
УМК факультета бизнес-коммуникаций
и информатики

Разработан в соответствии с ФГОС ВО

с учетом требований проф. стандарта

Председатель УМК

В.К. Карнаухова

ФИО, должность, ученая степень, звание

подпись, печать

Разработчики:

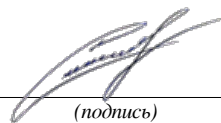
(подпись)

доцент

(занимаемая должность)

И.С. Петрушин

(инициалы, фамилия)


(подпись)

доцент

(занимаемая должность)

М.А. Сокольская

(инициалы, фамилия)

Цель фонда оценочных средств. Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «Б1.В.14 Основы мобильной разработки». Перечень видов оценочных средств соответствует рабочей программе дисциплины.

Фонд оценочных средств включает контрольные материалы для проведения текущего контроля (в следующих формах: тест, устный опрос, практическое задание, доклад-/презентация, конспект лекций) и промежуточной аттестации в форме вопросов и заданий к экзамену.

Структура и содержание заданий – задания разработаны в соответствии с рабочей программой дисциплины «Б1.В.14 Основы мобильной разработки».

1. Паспорт фонда оценочных средств

| Компетенция | Индикаторы компетенций | Результаты обучения |
|---|------------------------|--|
| ПК-2 Способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение | ПК-2.1 | Знать технологии разработки программного обеспечения: методы, средства, процедуры и инструменты |
| | ПК-2.2 | Уметь внедрять и адаптировать прикладное программное обеспечение |
| | ПК-2.3 | Владеть навыками решения задач реализации и модификации ПО: планирования и оценки проекта по разработке ПО; анализа системных и программных требований; проектирования алгоритмов, структур данных и программных структур; кодирования с использованием различных языков программирования и разметки; рефакторинга ПО; тестирования и отладки программного кода; сопровождения |
| ПК-3 Способность осуществлять тестирование компонентов информационных систем и по заданным сценариям | ПК-3.1 | Знать основные методы тестирования компонентов информационных систем |
| | ПК-3.2 | Уметь пользоваться специальным программным обеспечением для автоматизированного тестирования (при необходимости) на уровне запуска готовых тестов |
| | ПК-3.3 | Владеть навыками разработки тестовых сценариев компонентов информационных систем, проведения тестирования, исследования и анализа результатов |

| Компетенция | Индикаторы компетенций | Результаты обучения |
|---|------------------------|---|
| ПК-5 Способность разрабатывать графические элементы информационных систем и сервисов | ПК-5.1 | Знать функциональные возможности графических программ, специальных сред разработки и сервисов создания графических элементов и визуализаций |
| | ПК-5.2 | Уметь использовать пакеты графических программ для разработки элементов дизайн-контента информационных систем и сервисов, в том числе пользовательских интерфейсов |
| | ПК-5.3 | Владеть навыками проектирования и разработки графического контента для информационных систем и сервисов в соответствии с прикладной задачей разработки программного обеспечения |

2. Показатели и критерии оценивания компетенций, шкалы оценивания

2.1. Показатели и критерии оценивания компетенций

| № п/п | Раздел, тема | Код индикатора компетенции | Наименование ОС | |
|-------|--------------------------------------|--|-----------------|----------|
| | | | ТК | ПА |
| 1 | Синтаксис языка Kotlin | ПК-2.1, ПК-2.2, ПК-3.1, ПК-3.2, ПК-3.3, ПК-5.1, ПК-5.3, ПК-2.3, ПК-5.2 | Тест, Пз, Д | Тест, УО |
| 2 | ООП в Kotlin | ПК-2.1, ПК-2.2, ПК-5.2, ПК-3.1, ПК-3.3, ПК-2.3, ПК-5.1 | Тест, Пз, Д | Тест, УО |
| 3 | Принципы работы мобильных приложений | ПК-3.1, ПК-5.1, ПК-5.3, ПК-2.1, ПК-2.2, ПК-5.2, ПК-3.2 | Тест, КЛ, Пз | Тест, УО |
| 4 | Разработка интерфейса приложений | ПК-2.1, ПК-2.2, ПК-2.3, ПК-3.1, ПК-3.3, ПК-5.2, ПК-5.1, ПК-5.3, ПК-3.2 | Тест, Д, Пз | Тест, УО |

2.2. Критерии оценивания результатов обучения для текущего контроля успеваемости и промежуточной аттестации

| Оценочное средство | Критерии оценивания | Шкала оценивания |
|--------------------|---|---------------------|
| Тест | Студентом даны правильные ответы на 91-100% заданий | Отлично |
| | Студентом даны правильные ответы на 81-90% заданий | Хорошо |
| | Студентом даны правильные ответы на 71-80% заданий | Удовлетворительно |
| | Студентом даны правильные ответы менее чем на 70% заданий | Неудовлетворительно |

| Оценочное средство | Критерии оценивания | Шкала оценивания |
|---------------------------|---|-------------------------|
| Устный опрос | <p>Ответ соответствует поставленной теме и содержит ответы на поставленные задачи, имеет четкую структуру, логически сопоставляемую с поставленными вопросами. Ответ демонстрирует способность анализировать и обобщать информацию, опираясь на знания, полученные в ходе изучения темы, а также демонстрировать самостоятельность автора в решении поставленных задач. Ответ содержит качественную речь и аргументацию, которая убедительно подтверждает выводы и ответы на поставленные вопросы</p> | Отлично |
| | <p>Ответ должен быть направлен на ответ на поставленные вопросы и соответствовать поставленной теме, иметь логическую цепочку рассуждений и четко демонстрировать связь между поставленными вопросами. Ответ выдержан в четкой форме, быть грамотно и без ошибок озвучен, выделены ключевые термины. Ответ должен демонстрировать способность анализировать и критически оценивать информацию, выбирая ключевые аспекты и выделяя главные выводы</p> | Хорошо |
| | <p>Ответ должен соответствовать поставленной теме и содержать ответы на поставленные вопросы, должен содержать существенную информацию, ясно передавать ответы и идеи. Ответ должен содержать достаточное количество аргументов и примеров, связанных с темой работы и позволяющих изложить свою точку зрения. Ответ должен быть грамотно сформулирован</p> | Удовлетворительно |
| | <p>Ответ не соответствует поставленной теме или не содержит ответов на поставленные задачи, содержит недостаточно аргументации и примеров, которые подтверждают высказанные в ответе идеи и выводы. Ответ не соответствует логической цепочке рассуждений и не выполняет требования логической последовательности высказывания, затрудняющей понимание ответа. Ответ содержит грубые ошибки, что затрудняет понимание высказывания</p> | Неудовлетворительно |

| Оценочное средство | Критерии оценивания | Шкала оценивания |
|---------------------------|---|-------------------------|
| Практическое задание | Задание выполнено верно. Выбран оптимальный путь решения. Присутствует развернутое описание алгоритма решения | Отлично |
| | Задание выполнено верно. Допущены негрубые логические ошибки при описании алгоритма решения. Отсутствуют пояснения к решению задания | Хорошо |
| | Ход решения задания верный, но допущены ошибки приведшие к неправильному ответу | Удовлетворительно |
| | В работе получен неверный ответ, связанный с грубыми ошибками допущенными в ходе решения, либо решение отсутствует полностью | Неудовлетворительно |
| Доклад/презентация | Обучающийся демонстрирует исчерпывающее знание материала и последовательно, четко и логично излагает материал, хорошо ориентируется в материале, не затрудняется с ответом на сопутствующие вопросы | Отлично |
| | Обучающийся демонстрирует знание материала, не допускает существенных неточностей. Ответы на дополнительные вопросы в целом верные, но содержащие отдельные пробелы | Хорошо |
| | Обучающийся демонстрирует знания только основного материала, но не знает деталей, допускает неточности как в докладе, так и в ответах на вопросы | Удовлетворительно |
| | Обучающийся не знает значительной части программного материала, плохо ориентируется в материале, допускает существенные ошибки, выступает неуверенно, с большими затруднениями | Неудовлетворительно |

| Оценочное средство | Критерии оценивания | Шкала оценивания |
|---------------------------|---|-------------------------|
| Конспект лекций | Все темы, предложенные для конспектирования были проработаны обучающимся, прочитан материал источников, выбрано главное и второстепенное, установлена логическая связь между элементами темы, выделены ключевые слова и понятия, конспект написан лаконично с применением системы условных сокращений | Отлично |
| | Все темы, предложенные для конспектирования были проработаны обучающимся, прочитан материал источников, выбрано главное и второстепенное, установлена логическая связь между элементами темы, выделены ключевые слова и понятия, конспект написан лаконично с применением системы условных сокращений | Хорошо |
| | Текст конспекта оформлен аккуратно, обучающимся выбрано главное и второстепенное, выделены ключевые слова и понятия | Удовлетворительно |
| | Текст конспекта не соответствует теме или не отражает ключевых положений изучаемой темы | Неудовлетворительно |

2.3. Оценочные средства для текущего контроля (примеры)

2.3.1. Материалы для компьютерного тестирования обучающихся

Общие критерии оценивания

| Процент правильных ответов | Оценка |
|-----------------------------------|-------------------------|
| 91% – 100% | 5 (отлично) |
| 81% – 90% | 4 (хорошо) |
| 71% – 80% | 3 (удовлетворительно) |
| Менее 70% | 2 (неудовлетворительно) |

Соответствие вопросов теста индикаторам формируемых и оцениваемых компетенций

| № вопроса в тесте | Код индикатора компетенции |
|--------------------------|-----------------------------------|
| 1 | ПК-3.1, ПК-5.2, ПК-5.3 |
| 2 | ПК-3.3, ПК-5.1, ПК-5.3 |
| 3 | ПК-2.1, ПК-2.2, ПК-2.3 |
| 4 | ПК-2.1, ПК-2.2, ПК-5.2 |

| № вопроса в тесте | Код индикатора компетенции |
|-------------------|----------------------------|
| 5 | ПК-2.1, ПК-3.3, ПК-5.2 |
| 6 | ПК-2.1, ПК-3.1, ПК-5.3 |
| 7 | ПК-3.1, ПК-5.1, ПК-5.3 |
| 8 | ПК-3.1, ПК-3.2, ПК-5.1 |
| 9 | ПК-2.1, ПК-2.2, ПК-3.1 |
| 10 | ПК-2.2, ПК-3.2, ПК-3.3 |
| 11 | ПК-2.1, ПК-2.2, ПК-3.1 |
| 12 | ПК-2.1, ПК-3.1, ПК-5.1 |
| 13 | ПК-2.1, ПК-2.2, ПК-5.2 |
| 14 | ПК-2.1, ПК-3.1, ПК-3.3 |

Ключ ответов

| № вопроса в тесте | Номер ответа (или ответ, или соответствие) |
|-------------------|--|
| 1 | 1b, 2a |
| 2 | 1d, 2a, 3b |
| 3 | 1a, 2b, 3c |
| 4 | 50 ± 0 |
| 5 | 1b, 2c, 3a |
| 6 | 1e, 2a, 3c |
| 7 | 1c, 2b, 3a |
| 8 | 14 ± 0 |
| 9 | 1a, 2f, 3b, 4c, 5e, 6d |
| 10 | 1b, 2d, 3e, 4a, 5c |
| 11 | bar, "bar" |
| 12 | 1b, 2a |
| 13 | c, d |
| 14 | 1b, 2d, 3a |

Перечень тестовых вопросов

№ 1. Задание на соответствие. Соотнесите элементы двух списков.

Соотнесите методы класса View и их назначение

- | | |
|-------------------|-------------------------------------|
| 1. invalidate() | a. обработка касания |
| 2. onTouchEvent() | b. перерисовка элементов интерфейса |
| | c. отмена изменений |

№ 2. Задание на соответствие. Соотнесите элементы двух списков.

Каждому описанию деятельности подберите приложение Android-системы, выполняющее эту роль

- | | |
|--|--------------------|
| 1. Организация доступа к внутренним ресурсам приложения | a. LocationManager |
| 2. Отслеживание и передача приложениям местоположения устройства | b. ActivityManager |
| | c. WhereManager |
| 3. Контроль жизненного цикла активностей приложения | d. ResourceManager |

№ 3. Задание на последовательность. Расположите в правильном порядке.

Расположите методы работы активности по порядку их выполнения

- a. onStart()
- b. onStop()
- c. onDestroy()

№ 4. Задание открытой формы. Введите числовой ответ.

Квадрат с какой длиной стороны будет нарисован при вызове метода drawRect(10F, 10F, 60F, 60F, Paint())?

№ 5. Задание на последовательность. Расположите в правильном порядке.

Расположите уровни процессов операционной системы Android по возрастанию начиная с ядра ОС

- a. приложения
- b. ядро системы
- c. подключаемые библиотеки

№ 6. Задание на соответствие. Соотнесите элементы двух списков.

Сопоставьте каждый оператор/ключевое слово с их ролью в языке Kotlin

- | | |
|---|----------|
| 1. такого ключевого слова (оператора) нет | a. final |
| 2. запрет на переопределение функции | b. when |
| 3. обращение к экземпляру класса | c. this |
| | d. null |
| | e. none |

№ 7. Задание на последовательность. Расположите в правильном порядке.

Задан массив строк в файле strings.xml (см. ниже), каким будет идентификатор ресурса такого массива? Считайте, что элементы последовательности будут записаны через точку (т.е. A.B.C) <resources> <string-array name="planets_array"> <item>Mercury</item> <item>Venus</item> <item>Earth</item> <item>Mars</item> </string-array> </resources>

- a. planets_array
- b. array
- c. R

№ 8. Задание открытой формы. Введите числовой ответ.

Гипотеза Коллатца - известная нерешённая математическая задача. Рассмотрим функцию, которая чётное число делит пополам, а нечётное умножает на 3 и прибавляет единицу. Суть гипотезы в том, что для любого натурального числа $n > 1$ последовательное применение такой функции всегда приведёт к единице. Например: $8 \rightarrow 4 \rightarrow 2 \rightarrow 1$ (3 шага) или $3 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$ (7 шагов) Сколько шагов сделает цикл while ? var n = 11 var count = 0 while (n > 1) { count ++ n = if (n % 2 == 0) n / 2 else n*3 + 1 }

№ 9. Задание на последовательность. Расположите в правильном порядке.

Расположите фрагменты программы так, чтобы код соответствовал выводу на экран всплывающего сообщения с текстом "Hello"

- a. Toast.
- b. this,
- c. "Hello",
- d. show()
- e. Toast.LENGTH_SHORT).

f. makeText()

№ 10. Задание на последовательность. Расположите в правильном порядке.

Соберите простую программу вывода последней цифры введённого пользователем числа

- a. println(a % 10)
- b. fun main(){
- c. }
- d. val sc = Scanner(System.`in`)
- e. val a:Int = sc.nextInt()

№ 11. Задание открытой формы. Введите ответ.

В программе выполняются следующие действия со строковыми переменными var s1 = "foo" var s2 = "bar" var s3 = "buzz" s3 = s1 s1 = s2 s2 = s3 println(s1) Какое значение выведет программа?

№ 12. Задание на соответствие. Соотнесите элементы двух списков.

По какому принципу определяется размер элемента интерфейса для приведённых параметров?

- | | |
|-----------------|--|
| 1. match_parent | a. по содержимому |
| 2. wrap_content | b. пространство родительского элемента |
| | c. случайным образом |

№ 13. Задание с множественным выбором. Выберите 2 правильных ответа.

Выберите верные утверждения об ассоциативных массивах (словарях) в Kotlin

- a. Ключами могут быть только строковые значения
- b. Значениями могут быть только числовые значения
- c. Одинаковое значение может соответствовать нескольким ключам
- d. Ключи не могут повторяться

№ 14. Задание на соответствие. Соотнесите элементы двух списков.

Сопоставьте каждый оператор/ключевое слово с их ролью в языке Kotlin

- | | |
|---------------------------------------|----------|
| 1. разрешение наследования класса | a. this |
| 2. вызов конструктора базового класса | b. open |
| 3. обращение к экземпляру класса | c. when |
| | d. super |
| | e. null |

2.3.2. Доклады/презентации для оценки компетенции «ПК-2.1»

№ 1. Сравнение языка программирования Kotlin с распространёнными аналогами.

Подготовьте сравнительный анализ языка программирования Kotlin с другими языками высокого уровня (Java, Python, C++ и др.), основываясь на собственном опыте разработки и литературных данных

№ 2. Сравнение механизмов наследования в различных языках высокого уровня.

Подготовьте сравнительный анализ механизмов наследования языка программирования Kotlin с другими языками высокого уровня (Java, Python, C++ и др.), основываясь на собственном опыте разработки и литературных данных

№ 3. Проектирование интерфейса с использованием различных типов разметки.

Найдите в публичных репозиториях и подготовьте короткий доклад о сторонних

виджетах, которые могут пригодиться в индивидуальном проекте.

2.3.3. Доклады/презентации для оценки компетенции «ПК-2.2»

№ 4. Сравнение языка программирования Kotlin с распространёнными аналогами.

Подготовьте сравнительный анализ языка программирования Kotlin с другими языками высокого уровня (Java, Python, C++ и др.), основываясь на собственном опыте разработки и литературных данных

№ 5. Проектирование интерфейса с использованием различных типов разметки.

Найдите в публичных репозиториях и подготовьте короткий доклад о сторонних виджетах, которые могут пригодиться в индивидуальном проекте.

2.3.4. Доклады/презентации для оценки компетенции «ПК-3.3»

№ 6. Сравнение языка программирования Kotlin с распространёнными аналогами.

Подготовьте сравнительный анализ языка программирования Kotlin с другими языками высокого уровня (Java, Python, C++ и др.), основываясь на собственном опыте разработки и литературных данных

№ 7. Сравнение механизмов наследования в различных языках высокого уровня.

Подготовьте сравнительный анализ механизмов наследования языка программирования Kotlin с другими языками высокого уровня (Java, Python, C++ и др.), основываясь на собственном опыте разработки и литературных данных

2.3.5. Доклады/презентации для оценки компетенции «ПК-5.3»

№ 8. Сравнение языка программирования Kotlin с распространёнными аналогами.

Подготовьте сравнительный анализ языка программирования Kotlin с другими языками высокого уровня (Java, Python, C++ и др.), основываясь на собственном опыте разработки и литературных данных

2.3.6. Доклады/презентации для оценки компетенции «ПК-3.1»

№ 9. Сравнение механизмов наследования в различных языках высокого уровня.

Подготовьте сравнительный анализ механизмов наследования языка программирования Kotlin с другими языками высокого уровня (Java, Python, C++ и др.), основываясь на собственном опыте разработки и литературных данных

2.3.7. Доклады/презентации для оценки компетенции «ПК-5.1»

№ 10. Проектирование интерфейса с использованием различных типов разметки.

Найдите в публичных репозиториях и подготовьте короткий доклад о сторонних виджетах, которые могут пригодиться в индивидуальном проекте.

2.3.8. Конспекты лекций для оценки компетенции «ПК-2.2»

№ 11. Жизненный цикл приложения, архитектура ОС Android.

Подготовьте конспект основных этапов жизненного цикла приложения. Опишите, какие события происходят при повороте экрана устройства.

2.3.9. Конспекты лекций для оценки компетенции «ПК-2.3»

№ 12. Жизненный цикл приложения, архитектура ОС Android.

Подготовьте конспект основных этапов жизненного цикла приложения. Опишите, какие события происходят при повороте экрана устройства.

2.3.10. Конспекты лекций для оценки компетенции «ПК-5.3»

№ 13. Жизненный цикл приложения, архитектура ОС Android.

Подготовьте конспект основных этапов жизненного цикла приложения. Опишите, какие события происходят при повороте экрана устройства.

3. Промежуточная аттестация

3.1. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и опыта деятельности

Экзамен является заключительным этапом процесса формирования компетенций обучающегося при изучении дисциплины и имеет целью проверку и оценку знаний обучающегося по теории, и применению полученных знаний, умений и навыков при решении практических задач.

Экзамен проводится по расписанию, сформированному учебно-методическим управлением, в сроки, предусмотренные календарным учебным графиком. Экзамен принимается преподавателем, ведущим лекционные занятия.

Экзамен проводится только при предъявлении обучающимся зачетной книжки и при условии выполнения всех контрольных мероприятий, предусмотренных учебным планом и рабочей программой дисциплины. Обучающимся на экзамене представляется право выбрать один из билетов. Время подготовки к ответу составляет 30 минут. По истечении установленного времени обучающийся должен ответить на вопросы экзаменационного билета. Результаты экзамена оцениваются по четырехбалльной системе и заносятся в зачетно-экзаменационную ведомость и зачетную книжку. В зачетную книжку заносятся только положительные оценки. Подписанный преподавателем экземпляр ведомости сдается не позднее следующего дня в деканат.

В случае неявки обучающегося на экзамен в зачетно-экзаменационную ведомость делается отметка «не явка». Обучающиеся, не прошедшие промежуточную аттестацию по дисциплине, должны ликвидировать академическую задолженность в установленном локальными нормативными актами порядке.

3.2. Вопросы к экзамену

| № | Вопрос | Код компетенции |
|----|---|------------------------|
| 1. | Каким образом правильно объявить переменную целочисленного типа, изменяющую впоследствии свои значения? | ПК-3.1, ПК-3.3, ПК-5.3 |
| 2. | Каким способом можно вывести на экран значение целочисленной переменной a ? Перечислите все допустимые варианты. | ПК-2.1, ПК-2.2, ПК-3.1 |
| 3. | Приведите примеры преимуществ использования наследования при разработке интерфейса мобильного приложения | ПК-2.1, ПК-3.1, ПК-3.2 |
| 4. | Поясните, для чего может потребоваться ограничение доступа к полям и методам класса | ПК-2.1, ПК-3.1, ПК-5.3 |
| 5. | Приведите примеры использования различных вариантов разметки: линейной, относительной и пр. Какие преимущества и ограничения имеют эти типы разметок? | ПК-2.1, ПК-3.2, ПК-5.3 |
| 6. | Опишите, какие события возникают в приложении во время его жизненного цикла | ПК-2.1, ПК-2.2, ПК-3.1 |

| № | Вопрос | Код компетенции |
|----|---|------------------------|
| 7. | Перечислите, какие методы требуется определить при разработке собственного элемента интерфейса на основе класса View | ПК-2.1, ПК-2.3, ПК-5.1 |
| 8. | Какие типы ресурсов предусмотрены в мобильном приложении? Укажите, в каких разделах располагаются XML-файлы с цветами, разметкой, строками и доступными разрешениями. | ПК-2.1, ПК-2.2, ПК-3.2 |

3.3. Тематика курсовых работ

По данной дисциплине выполнение курсовых проектов (работ) не предусматривается.

3.4. Материалы для компьютерного тестирования обучающихся

Общие критерии оценивания

| Процент правильных ответов | Оценка |
|----------------------------|-------------------------|
| 91% – 100% | 5 (отлично) |
| 81% – 90% | 4 (хорошо) |
| 71% – 80% | 3 (удовлетворительно) |
| Менее 70% | 2 (неудовлетворительно) |

Соответствие вопросов теста индикаторам формируемых и оцениваемых компетенций

| № вопроса в тесте | Код индикатора компетенции |
|-------------------|----------------------------|
| 1 | ПК-3.1, ПК-5.2, ПК-5.3 |
| 2 | ПК-3.3, ПК-5.1, ПК-5.3 |
| 3 | ПК-2.1, ПК-2.2, ПК-2.3 |
| 4 | ПК-2.1, ПК-2.2, ПК-5.2 |
| 5 | ПК-2.1, ПК-3.3, ПК-5.2 |
| 6 | ПК-2.1, ПК-3.1, ПК-5.3 |
| 7 | ПК-3.1, ПК-5.1, ПК-5.3 |
| 8 | ПК-3.1, ПК-3.2, ПК-5.1 |
| 9 | ПК-2.1, ПК-2.2, ПК-3.1 |
| 10 | ПК-2.2, ПК-3.2, ПК-3.3 |
| 11 | ПК-2.1, ПК-2.2, ПК-3.1 |
| 12 | ПК-2.1, ПК-3.1, ПК-5.1 |
| 13 | ПК-2.1, ПК-2.2, ПК-5.2 |
| 14 | ПК-2.1, ПК-3.1, ПК-3.3 |

Ключ ответов

| № вопроса в тесте | Номер ответа (или ответ, или соответствие) |
|-------------------|--|
| 1 | 1b, 2a |
| 2 | 1d, 2a, 3b |
| 3 | 1a, 2b, 3c |

| № вопроса в тесте | Номер ответа (или ответ, или соответствие) |
|-------------------|--|
| 4 | 50 ± 0 |
| 5 | 1b, 2c, 3a |
| 6 | 1e, 2a, 3c |
| 7 | 1c, 2b, 3a |
| 8 | 14 ± 0 |
| 9 | 1a, 2f, 3b, 4c, 5e, 6d |
| 10 | 1b, 2d, 3e, 4a, 5c |
| 11 | bar, "bar" |
| 12 | 1b, 2a |
| 13 | c, d |
| 14 | 1b, 2d, 3a |

Перечень тестовых вопросов

№ 1. Задание на соответствие. Соотнесите элементы двух списков.

Соотнесите методы класса View и их назначение

- | | |
|-------------------|-------------------------------------|
| 1. invalidate() | a. обработка касания |
| 2. onTouchEvent() | b. перерисовка элементов интерфейса |
| | c. отмена изменений |

№ 2. Задание на соответствие. Соотнесите элементы двух списков.

Каждому описанию деятельности подберите приложение Android-системы, выполняющее эту роль

- | | |
|--|--------------------|
| 1. Организация доступа к внутренним ресурсам приложения | a. LocationManager |
| 2. Отслеживание и передача приложениям местоположения устройства | b. ActivityManager |
| 3. Контроль жизненного цикла активностей приложения | c. WhereManager |
| | d. ResourceManager |

№ 3. Задание на последовательность. Расположите в правильном порядке.

Расположите методы работы активности по порядку их выполнения

- onStart()
- onStop()
- onDestroy()

№ 4. Задание открытой формы. Введите числовой ответ.

Квадрат с какой длиной стороны будет нарисован при вызове метода drawRect(10F, 10F, 60F, 60F, Paint())?

№ 5. Задание на последовательность. Расположите в правильном порядке.

Расположите уровни процессов операционной системы Android по возрастанию начиная с ядра ОС

- приложения
- ядро системы
- подключаемые библиотеки

№ 6. Задание на соответствие. Соотнесите элементы двух списков.

Сопоставьте каждый оператор/ключевое слово с их ролью в языке Kotlin

- | | |
|---|----------|
| 1. такого ключевого слова (оператора) нет | a. final |
| 2. запрет на переопределение функции | b. when |
| 3. обращение к экземпляру класса | c. this |
| | d. null |
| | e. none |

№ 7. Задание на последовательность. Расположите в правильном порядке.

Задан массив строк в файле strings.xml (см. ниже), каким будет идентификатор ресурса такого массива? Считайте, что элементы последовательности будут записаны через точку (т.е. A.B.C) `<resources> <string-array name="planets_array"> <item>Mercury</item> <item>Venus</item> <item>Earth</item> <item>Mars</item> </string-array> </resources>`

- a. planets_array
- b. array
- c. R

№ 8. Задание открытой формы. Введите числовой ответ.

Гипотеза Коллатца - известная нерешённая математическая задача. Рассмотрим функцию, которая чётное число делит пополам, а нечётное умножает на 3 и прибавляет единицу. Суть гипотезы в том, что для любого натурального числа $n > 1$ последовательное применение такой функции всегда приведёт к единице. Например: $8 \rightarrow 4 \rightarrow 2 \rightarrow 1$ (3 шага) или $3 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$ (7 шагов) Сколько шагов сделает цикл while ? `var n = 11 var count = 0 while (n > 1) { count ++ n = if (n % 2 == 0) n / 2 else n*3 + 1 }`

№ 9. Задание на последовательность. Расположите в правильном порядке.

Расположите фрагменты программы так, чтобы код соответствовал выводу на экран всплывающего сообщения с текстом "Hello"

- a. Toast.
- b. this,
- c. "Hello",
- d. show()
- e. Toast.LENGTH_SHORT).
- f. makeText(

№ 10. Задание на последовательность. Расположите в правильном порядке.

Соберите простую программу вывода последней цифры введённого пользователем числа

- a. println(a % 10)
- b. fun main(){
- c. }
- d. val sc = Scanner(System.`in`)
- e. val a:Int = sc.nextInt()

№ 11. Задание открытой формы. Введите ответ.

В программе выполняются следующие действия со строковыми переменными `var s1 = "foo" var s2 = "bar" var s3 = "buzz" s3 = s1 s1 = s2 s2 = s3 println(s1)` Какое значение выведет программа?

№ 12. Задание на соответствие. Соотнесите элементы двух списков.

По какому принципу определяется размер элемента интерфейса для приведённых параметров?

- | | |
|-----------------|--|
| 1. match_parent | a. по содержимому |
| 2. wrap_content | b. пространство родительского элемента |
| | c. случайным образом |

№ 13. Задание с множественным выбором. Выберите 2 правильных ответа.

Выберите верные утверждения об ассоциативных массивах (словарях) в Kotlin

- a. Ключами могут быть только строковые значения
- b. Значениями могут быть только числовые значения
- c. Одинаковое значение может соответствовать нескольким ключам
- d. Ключи не могут повторяться

№ 14. Задание на соответствие. Соотнесите элементы двух списков.

Сопоставьте каждый оператор/ключевое слово с их ролью в языке Kotlin

- | | |
|---------------------------------------|----------|
| 1. разрешение наследования класса | a. this |
| 2. вызов конструктора базового класса | b. open |
| 3. обращение к экземпляру класса | c. when |
| | d. super |
| | e. null |

3.5. Вопросы для коллоквиумов, собеседования для оценки компетенции

«ПК-2.1»

№ 1. Типы данных и операторов в Kotlin.

Приведите примеры применения операторов when, apply, let, а также Elvis-оператора.

№ 2. Применение ООП.

Приведите примеры применения механизмов наследования при разработке элементов интерфейса мобильных приложений

№ 3. Приёмы проектирования интерфейса.

Опишите возможности различных вариантов верстки интерфейса. Как выполняется обработка событий, например, нажатие на кнопку.

№ 4. Верстка с помощью XML и динамическое создание элементов интерфейса.

Приведите примеры, когда необходимо создавать собственный элемент интерфейса. Сравните с использованием готовой статической разметки в XML.

3.6. Вопросы для коллоквиумов, собеседования для оценки компетенции

«ПК-2.2»

№ 5. Типы данных и операторов в Kotlin.

Приведите примеры применения операторов when, apply, let, а также Elvis-оператора.

№ 6. Верстка с помощью XML и динамическое создание элементов интерфейса.

Приведите примеры, когда необходимо создавать собственный элемент интерфейса. Сравните с использованием готовой статической разметки в XML.

3.7. Вопросы для коллоквиумов, собеседования для оценки компетенции

«ПК-5.1»

№ 7. Типы данных и операторов в Kotlin.

Приведите примеры применения операторов when, apply, let, а также Elvis-оператора.

**3.8. Вопросы для коллоквиумов, собеседования для оценки компетенции
«ПК-3.1»**

№ 8. Применение ООП.

Приведите примеры применения механизмов наследования при разработке элементов интерфейса мобильных приложений

**3.9. Вопросы для коллоквиумов, собеседования для оценки компетенции
«ПК-5.2»**

№ 9. Применение ООП.

Приведите примеры применения механизмов наследования при разработке элементов интерфейса мобильных приложений

**3.10. Вопросы для коллоквиумов, собеседования для оценки компетенции
«ПК-3.3»**

№ 10. Приёмы проектирования интерфейса.

Опишите возможности различных вариантов вёрстки интерфейса. Как выполняется обработка событий, например, нажатие на кнопку.

№ 11. Вёрстка с помощью XML и динамическое создание элементов интерфейса.

Приведите примеры, когда необходимо создавать собственный элемент интерфейса. Сравните с использованием готовой статической разметки в XML.

**3.11. Вопросы для коллоквиумов, собеседования для оценки компетенции
«ПК-5.3»**

№ 12. Приёмы проектирования интерфейса.

Опишите возможности различных вариантов вёрстки интерфейса. Как выполняется обработка событий, например, нажатие на кнопку.