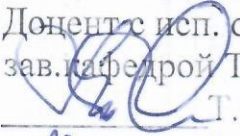


Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного
учреждения высшего образования
«Пермский национальный исследовательский политехнический университет»

УТВЕРЖДАЮ

Доцент с исп. обязанностей
зав. кафедрой ТД

 Т.О. Сопина

« 28 » 02 2025 г

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

**для проведения текущего контроля успеваемости и промежуточной
аттестации обучающихся по профессиональному модулю**

ПМ.01 РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ

Приложение к рабочей программе профессионального модуля

основной профессиональной образовательной программы
подготовки специалистов среднего звена
по специальности СПО 09.02.07 Информационные системы и программирование
(базовая подготовка)

Лысьва, 2025

Фонд оценочных средств разработан на основе:

— Федерального государственного образовательного стандарта среднего профессионального образования, утверждённого приказом Министерства образования и науки Российской Федерации «09» декабря 2016 г. № 1547 по специальности 09.02.07 Информационные системы и программирование;

— Рабочей программы профессионального модуля ПМ.01 Разработка модулей программного обеспечения для компьютерных систем, утвержденной «28» 02 2025 г.

Разработчики:

преподаватель высшей категории С.А. Зыкин

преподаватель М.Н. Апталаев



Фонд оценочных средств рассмотрен и одобрен на заседании предметной (цикловой) комиссии *Естественнонаучных дисциплин* (ПЦК ЕНД) «11» февраля 2025 г., протокол №06

Председатель ПЦК ЕНД



М.Н. Апталаев

Генеральный директор
ООО "ИПК "Техноконтроль"



И.В. Сошин

ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

1 Область применения

ФОС предназначен для проверки результатов освоения профессионального **ПМ.01 Разработка модулей программного обеспечения для компьютерных систем** по специальности СПО *09.02.07 Информационные системы и программирование* в части овладения видом профессиональной деятельности «*Разработка модулей программного обеспечения для компьютерных систем*».

В результате освоения профессионального модуля обучающийся должен обладать предусмотренными ФГОС по специальности СПО *09.02.07 Информационные системы и программирование* следующими общими и профессиональными компетенциями.

Перечень общих компетенций элементы, которых формируются в рамках ПМ:

Код	Наименование общих компетенций
ОК 01	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;
ОК 03	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях;
ОК 04	Эффективно взаимодействовать и работать в коллективе и команде;
ОК 05	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста;
ОК 06	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения.
ОК 07	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях.
ОК 08	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
ОК 09	Пользоваться профессиональной документацией на государственном и иностранном языках.

Перечень профессиональных компетенций элементы, которых формируются в рамках ПМ:

Код	Наименование видов деятельности и профессиональных компетенций
ВД 1	Разработка модулей программного обеспечения для компьютерных систем
ПК 1.1	Формировать алгоритмы разработки программных модулей в соответствии с

	техническим заданием
ПК 1.2	Разрабатывать программные модули в соответствии с техническим заданием
ПК 1.3	Выполнять отладку программных модулей с использованием специализированных программных средств
ПК 14	Выполнять тестирование программных модулей
ПК 1.5	Осуществлять рефакторинг и оптимизацию программного кода
ПК 1.6	Разрабатывать модули программного обеспечения для мобильных платформ

В результате освоения профессионального модуля обучающийся должен:

иметь практический опыт в:	– В разработке кода программного продукта на основе готовой спецификации на уровне модуля; – использовании инструментальных средств на этапе отладки программного продукта; – проведении тестирования программного модуля по определенному сценарию; – использовании инструментальных средств на этапе отладки программного продукта; – разработке мобильных приложений
уметь:	– осуществлять разработку кода программного модуля на языках низкого и высокого уровней; – создавать программу по разработанному алгоритму как отдельный модуль; – выполнять отладку и тестирование программы на уровне модуля; – осуществлять разработку кода программного модуля на современных языках программирования; – уметь выполнять оптимизацию и рефакторинг программного кода; – оформлять документацию на программные средства
иметь практический опыт в:	– В разработке кода программного продукта на основе готовой спецификации на уровне модуля; – использовании инструментальных средств на этапе отладки программного продукта; – проведении тестирования программного модуля по определенному сценарию; – использовании инструментальных средств на этапе отладки программного продукта; – разработке мобильных приложений

Перечень личностных результатов, которые формируются в рамках ПМ:

Код	Наименование личностных результатов
<i>ЛР 5</i>	Демонстрирующий умение эффективно взаимодействовать в команде, вести диалог, в том числе с использованием средств коммуникации
<i>ЛР 6</i>	Демонстрирующий навыки анализа и интерпретации информации из различных источников с учетом нормативно-правовых норм
<i>ЛР 7</i>	Демонстрирующий готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности.
<i>ЛР 8</i>	Пользоваться профессиональной документацией на государственном и иностранном языках.
<i>ЛР 9</i>	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения
<i>ЛР 10</i>	Использовать знания по финансовой грамотности, планировать предпринимательскую деятельность в профессиональной сфере.
<i>ЛР 11</i>	Планировать и реализовывать собственное профессиональное и личностное развитие в условиях развития информационных технологий, применяемых в различных отраслях народного хозяйства.
<i>ЛР 12</i>	Активно применяющий полученные знания на практике
<i>ЛР 13</i>	Способный анализировать производственную ситуацию, быстро принимать решения
<i>ЛР 14</i>	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
<i>ЛР 17</i>	Проявлять доброжелательность к окружающим, деликатность, чувство такта и готовность оказать услугу каждому кто в ней нуждается.

1 МЕТОДЫ И ФОРМЫ КОНТРОЛЯ ОЦЕНИВАНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Таблица 1 – Методы и формы контроля и оценивания элементов профессионального модуля

Элемент профессионального модуля	Методы и формы контроля и оценивания		
	Текущий контроль	Рубежный контроль	Промежуточная аттестация
МДК 01.01 Разработка программных модулей	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	Тестирование Курсовая работа	Другая форма контроля (4 семестр) Экзамен (5 семестр) Дифференцированный зачет (6 семестр) Экзамен (7 семестр)
МДК 01.02 Поддержка и тестирование программных модулей	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	Тестирование	Дифференцированный зачет (6 семестр) Экзамен (7 семестр)
МДК 01.03 Разработка мобильных приложений	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка	Тестирование Курсовая работа	Экзамен (5 семестр)

	результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ		
МДК 01.04 Системное программирование	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	Тестирование	Дифференцированный зачет (6 семестр) Экзамен (7 семестр)
УП 01.01 Учебная практика *	Экспертное наблюдение и оценка при выполнении работ на учебной практике Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной практики	-	Дифференцированный зачет (5 семестр)
ПП 01.01 Производственная практика (по профилю специальности)*	Экспертное наблюдение и оценка при выполнении работ на производственной практике (по профилю специальности) Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения производственной	-	Дифференцированный зачет (7 семестр)

	<i>практики (по профилю специальности)</i>		
ПМ 04. ЭК Экзамен по модулю	-	Дифзачет по МДК Экзамен по МДК Курсовая работа Дифференцированный зачет по учебной практике Дифференцированный зачет по производственной практике (по профилю специальности)	Экзамен по модулю (7 семестр)

**ФОС Учебной практики и Производственной практики (по профилю специальности) приведены отдельными документами*

2 КРИТЕРИИ ОЦЕНИВАНИЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ

Критерии оценки устного ответа

Критерии оценки	Оценка
обучающийся полно излагает материал (отвечает на вопрос), дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; излагает материал последовательно и правильно с точки зрения норм литературного языка	Отлично
обучающийся дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но допускает 1–2 ошибки, которые сам же исправляет, и 1–2 недочета в последовательности и языковом оформлении излагаемого	Хорошо
обучающийся обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого	Удовлетворительно
обучающийся обнаруживает незнание большей части соответствующего вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал. Оценка «2» отмечает такие недостатки в подготовке, которые являются серьезным препятствием к успешному овладению последующим материалом	Неудовлетворительно

Критерии оценки практических и лабораторных занятий

1 активность работы на практическом занятии (выполнение всех заданий, предложенных преподавателем);

2 правильность ответов на вопросы (верное, четкое и достаточно глубокое изложение понятий, идей, и т.д.);

3 полнота и одновременно лаконичность ответа (ответ должен отражать основные теории и концепции по раскрываемому вопросу, содержать их критический анализ и сопоставление);

4 умение формулировать собственную точку зрения, грамотно аргументировать свою позицию по раскрываемому вопросу;

5 культура речи (материал должен быть изложен хорошим профессиональным языком, с грамотным использованием соответствующей системы понятий и терминов)

Критерии оценки практического задания

Критерии оценки	Оценка
– практическое задание выполнено в установленный срок с использованием рекомендаций преподавателя – показан высокий уровень знания изученного материала по заданной теме – проявлен творческий подход – умение глубоко анализировать проблему и делать обобщающие практико-ориентированные выводы – работа выполнена без ошибок и недочетов или допущено не более одного недочета	Отлично
– практическое задание выполнено в установленный срок с использованием рекомендаций преподавателя – показан хороший уровень владения изученным материалом по заданной теме – работа выполнена полностью, но допущено в ней: а) не более одной негрубой ошибки и одного недочета б) или не более двух недочетов	Хорошо
– практическое задание выполнено в установленный срок с частичным использованием рекомендаций преподавателя – продемонстрированы минимальные знания по основным темам изученного материала – выполнено не менее половины работы или допущены в ней: а) не более двух грубых ошибок; б) не более одной грубой ошибки и одного недочета; в) не более двух-трех негрубых ошибок; г) одна негрубая ошибка и три недочета; д) при отсутствии ошибок, 4-5 недочетов	Удовлетворительно
– число ошибок и недочетов превосходит норму, при которой может быть выставлена оценка «удовлетворительно» или если правильно выполнено менее половины задания – если обучающийся не приступал к выполнению задания или правильно выполнил не более 10 процентов всех заданий	Неудовлетворительно

Критерии оценки лабораторного задания

Критерии оценки	Оценка
– работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов и измерений; все опыты проведены в условиях и режимах, обеспечивающих получение правильных результатов и выводов; соблюдены требования правил безопасности труда; в отчете правильно и аккуратно выполнены все записи, таблицы, рисунки, чертежи, графики, вычисления; правильно выполнен анализ погрешностей	Отлично
– работа выполнена в полном объеме с соблюдением необходимой последовательности проведения опытов и измерений; все опыты проведены в условиях и режимах, обеспечивающих получение правильных результатов и выводов; соблюдены требования правил безопасности труда; в отчете правильно и аккуратно выполнены все записи, таблицы, рисунки, чертежи, графики, вычисления; правильно выполнен анализ погрешностей	Хорошо

Допущено два - три недочета или не более одной негрубой ошибки и одного недочёта	
– работа выполнена не полностью, но объем выполненной части позволяет получить правильные результаты и выводы, если в ходе проведения опыта и измерений были допущены ошибки.	Удовлетворительно
Работа выполнена не полностью и объем выполненной части работы не позволяет сделать правильных выводов, если опыты, измерения, вычисления, наблюдения производились неправильно	Неудовлетворительно

Критерии оценки тестов

Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
86 - 100	70 - 85	51 - 69	50 и менее

Критерии результатов самостоятельной работы

При экспертной оценке результатов самостоятельной работы учитываются такие критерии:

- Глубина освоения знаний
- Источники информации
- Качество выполнения работы
- Самостоятельность изложения
- Творчество и личный вклад
- Соблюдение правил оформления

Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения профессионального модуля (ПМ)

Интегральная качественная оценка освоения профессионального модуля, в включая междисциплинарные курсы, учитываемая при промежуточной аттестации.

Экспертное наблюдение и оценка при выполнении работ на учебной и/или производственной практике (по профилю специальности)

Учебная практика направлена на формирование у обучающихся практических профессиональных умений, приобретение первоначального практического опыта в рамках модулей по основным видам профессиональной деятельности.

Производственная практика (по профилю специальности) направлена на приобретение опыта самостоятельной профессиональной деятельности в рамках модулей по основным видам профессиональной деятельности.

Текущий контроль результатов прохождения учебной и/или производственной (по профилю специальности) практики в соответствии с рабочей программой практики происходит при использовании следующих обязательных форм контроля:

- ежедневный контроль посещаемости практики;
- наблюдение за выполнением видов работ на практике;
- контроль качества выполнения видов работ на практике
- контроль за ведением дневника практики,
- контроль сбора материала для отчета по практике в соответствии с заданием на практику.

Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной и/ или производственной практики (по профилю специальности)

Интегральная качественная оценка освоения учебной и/или производственной практики (по профилю специальности), учитываемая при промежуточной аттестации по профессиональному модулю

3 КРИТЕРИИ ОЦЕНИВАНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Профессиональный модуль *ПМ.01 Разработка модулей программного обеспечения для компьютерных систем* изучается в течение 1 семестра.

Формами контроля промежуточной аттестации являются:

1 МДК 01.01 Разработка программных модулей:

- Другая форма контроля – 4 семестр;
- Экзамен – 5 семестр;
- Дифференцированный зачет – 6 семестр;
- Экзамен – 7 семестр.

2 МДК 01.02 Поддержка и тестирование программных модулей:

- Дифференцированный зачет – 6 семестр;
- Экзамен – 7 семестр.

3 МДК 01.03 Разработка мобильных приложений:

- Экзамен – 5 семестр.

4 МДК 01.04 Системное программирование:

- Дифференцированный зачет – 6 семестр;
- Экзамен – 7 семестр.

5 Учебная практика:

- дифференцированный зачет – 5 семестр;

6 Производственная практика (по профилю специальности):

- дифференцированный зачет - 7 семестр

7 Экзамен по модулю – 7 семестр

Критерии оценивания дифференцированного зачета (МДК)

Критерии оценки	Оценка
Всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполненные все предусмотренные программой задания, глубоко усвоенные основная и дополнительная литература, рекомендованная программой, активная работа на практических занятиях Обучающийся разбирается в основных научных концепциях по изучаемой учебной дисциплине, проявляет творческие способности и научный подход в понимании и изложении учебного программного материала Ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично	Отлично

Достаточно полное знание учебно-программного материала Обучающийся не допускает в ответе существенных неточностей, самостоятельно выполнил все предусмотренные программой задания, усвоил основную литературу, рекомендованную программой, активно работал на практических занятиях, показал систематический характер знаний по учебной дисциплине, достаточный для дальнейшей учёбы, а также способность к их самостоятельному пополнению	Хорошо
Обучающийся показал знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по специальности, не отличался активностью на практических занятиях, самостоятельно выполнил основные предусмотренные программой задания, однако допустил погрешности при их выполнении и в ответе на дифференцированном зачёте, но обладает необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей	Удовлетворительно
Обучающийся обнаруживает пробелы в знаниях или отсутствие знаний по значительной части основного учебно-программного материала, не выполнил самостоятельно предусмотренные программой основные задания, допустил принципиальные ошибки в выполнении предусмотренных программой заданий, не отработал основные практические занятия, допускает существенные ошибки при ответе и не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по соответствующей учебной дисциплине	Неудовлетворительно

Критерии оценивания экзамена (МДК)

Критерии оценки	Оценка
Всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполненные все предусмотренные программой задания, глубоко усвоенные основная и дополнительная литература, рекомендованная программой, активная работа на практических занятиях Обучающийся разбирается в основных научных концепциях по изучаемой учебной дисциплине, проявляет творческие способности и научный подход в понимании и изложении учебного программного материала Ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично	Отлично
Достаточно полное знание учебно-программного материала Обучающийся не допускает в ответе существенных неточностей, самостоятельно выполнил все предусмотренные программой задания, усвоил основную литературу, рекомендованную программой, активно работал на практических занятиях, показал систематический характер знаний по учебной дисциплине, достаточный для дальнейшей учёбы, а также способность к их самостоятельному пополнению	Хорошо

Обучающийся показал знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по специальности, не отличался активностью на практических занятиях, самостоятельно выполнил основные предусмотренные программой задания, однако допустил погрешности при их выполнении и в ответе на дифференцированном зачёте, но обладает необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей	Удовлетворительно
Обучающийся обнаруживает пробелы в знаниях или отсутствие знаний по значительной части основного учебно-программного материала, не выполнил самостоятельно предусмотренные программой основные задания, допустил принципиальные ошибки в выполнении предусмотренных программой заданий, не отработал основные практические занятия, допускает существенные ошибки при ответе и не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по соответствующей учебной дисциплине	Неудовлетворительно

Критерии оценивания дифференцированного зачета учебной и/или производственной практики (по профилю специальности)

Оценка качества прохождения практики происходит по следующим показателям:

- оценка в аттестационном листе уровня освоения профессиональных компетенций при выполнении работ на практике;
- запись в характеристике об освоении общих компетенций при выполнении работ на практике;

Оценка за дифференцированный зачет по практике определяется как средний балл за представленные материалы с практики и защиты отчета по практике. Оценка выставляется по 4-х балльной шкале.

Критерии оценивания результатов практики

Критерии оценки	Оценка
Комплект документов полный, все документы подписаны и заверены должным образом. Цель практики выполнена полностью или сверх того: полноценно отработаны и применены на практике три и более профессиональные компетенции (представлены многочисленные примеры и результаты деятельности). Замечания от организации (базы практики) отсутствуют, а работа обучающегося оценена на «отлично». Обучающийся аргументированно и убедительно прокомментировал отчет по практике. Отчет по практике представлен в срок, оформлен в соответствии с требованиями ГОСТ «ГОСТ 7.32-2017. Межгосударственный стандарт. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления», что свидетельствует о полной сформированности у обучающихся надлежащих компетенции	Отлично
Комплект документов полный, но некоторые документы не подписаны или заверены недолжным образом. Цель практики выполнена почти полностью: частично отработаны и применены на практике три и менее	Хорошо

профессиональные компетенции (кратко представлены некоторые примеры и результаты деятельности). Незначительные замечания от представителей организации (базы практики), а работа обучающегося оценена на «хорошо». Обучающийся убедительно и уверенно прокомментировал отчет по практике. Отчет по практике представлен в срок, однако имеются несущественные замечания в оформлении отчета, что свидетельствует о сформированности у обучающегося неявно выраженных надлежащих компетенций	
Комплект документов полный, но некоторые документы не подписаны или заверены недолжным образом. Цель практики выполнена частично: недостаточно отработаны и применены на практике три и менее профессиональные компетенции (кратко представлены некоторые примеры и результаты деятельности). Высказаны критические замечания от представителей организации (базы практики), а работа обучающегося оценена на «удовлетворительно». Обучающийся отвечал неполно, неуверенно прокомментировал отчет по практике. Отчет по практике представлен в срок, однако имеются существенные замечания по оформлению отчета, что свидетельствует о недостаточной сформированности у обучающегося надлежащих компетенций	Удовлетворительно
Комплект документов неполный. Цель практики выполнена эпизодически: не отработаны или некачественно применены на практике профессиональные компетенции (примеры и результаты деятельности отсутствуют). Высказаны серьезные замечания от представителей организации (базы практики), а работа обучающегося оценена на «неудовлетворительно». Обучающийся удовлетворительно не ответил на вопросы на экзамене. Отчет по практике представлен в срок, однако является неполным и не соответствует стандарту подготовки, что свидетельствует о несформированности у обучающегося надлежащих компетенций. Обучающийся практику не прошел по неуважительной причине. Обучающийся не представил отчетных документов	Неудовлетворительно

Критерии оценивания экзамена по модулю

Экзамен по модулю представляет собой форму независимой оценки результатов обучения с участием работодателей и проводится по завершении изучения учебной программы профессионального модуля.

Экзамен по модулю проверяет готовность обучающегося к выполнению указанного вида профессиональной деятельности и сформированности у него компетенций.

Итогом экзамена является однозначное решение: «вид профессиональной деятельности освоен / не освоен», профессиональный модуль: зачтено/не зачтено

Условием положительной аттестации «вид профессиональной деятельности освоен» является положительная оценка освоения всех профессиональных компетенций. При отрицательном заключении хотя бы по одной из профессиональных компетенций принимается решение «вид профессиональной деятельности не освоен».

Методы, критерии оценивания и условия проведения экзамена по модулю определяются индивидуально для каждого профессионального модуля.

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного учреждения
высшего образования
«Пермский национальный исследовательский политехнический университет»

ОЦЕНОЧНЫЕ СРЕДСТВА

МЕЖДИСЦИПЛИНАРНОГО КУРСА

МДК 01.01 Разработка программных модулей

основной профессиональной образовательной программы
подготовки специалистов среднего звена
по специальности СПО 09.02.07 *Информационные системы и программирование*
(базовая подготовка)

Лысьва, 2023

ОБЩИЕ ПОЛОЖЕНИЯ

Контрольно – измерительные материалы по результатам изучения междисциплинарного курса: *МДК 01.01 Разработка программных модулей* ориентированы на проверку степени достижения требований к минимуму содержания и уровню подготовки обучающихся в соответствии с ФГОС и является основополагающим документом для организации контроля знаний, умений, практического опыта обучающихся в учебном процессе.

Результатом освоения междисциплинарного курса *МДК 01.01 Разработка программных модулей* является:

Код и наименование профессиональных и общих компетенций, личностных результатов, формируемых в рамках ПМ	Основные показатели оценки результата
ПК 1.1 <i>Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием</i>	<i>Уметь:</i> — осуществлять разработку кода программного модуля на языках низкого и высокого уровней; — создавать программу по разработанному алгоритму как отдельный модуль. <i>Знать:</i> — основные этапы разработки программного обеспечения; — основные принципы технологии структурного и объектно-ориентированного программирования.
ПК 1.2 <i>Разрабатывать программные модули в соответствии с техническим заданием</i>	<i>Уметь:</i> — осуществлять разработку кода программного модуля на языках низкого и высокого уровней; — создавать программу по разработанному алгоритму как отдельный модуль. <i>Знать:</i> — основные этапы разработки программного обеспечения; — основные принципы технологии структурного и объектно-ориентированного программирования.
ЛР 5	Демонстрирующий умение эффективно взаимодействовать в команде, вести диалог, в том числе с использованием средств коммуникации
ЛР 6	Демонстрирующий навыки анализа и интерпретации информации из различных источников с учетом нормативно-правовых норм
ЛР 7	Демонстрирующий готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности.
ЛР 8	Пользоваться профессиональной документацией на государственном и иностранном языках.

<i>ЛР 9</i>	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения
<i>ЛР 10</i>	Использовать знания по финансовой грамотности, планировать предпринимательскую деятельность в профессиональной сфере.
<i>ЛР 11</i>	Планировать и реализовывать собственное профессиональное и личностное развитие в условиях развития информационных технологий, применяемых в различных отраслях народного хозяйства.
<i>ЛР 12</i>	Активно применяющий полученные знания на практике
<i>ЛР 13</i>	Способный анализировать производственную ситуацию, быстро принимать решения
<i>ЛР 14</i>	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
<i>ЛР 17</i>	Проявлять доброжелательность к окружающим, деликатность, чувство такта и готовность оказать услугу каждому кто в ней нуждается.

1 МЕТОДЫ И ФОРМЫ КОНТРОЛЯ ОЦЕНИВАНИЯ ЭЛЕМЕНТОВ МЕЖДИСЦИПЛИНАРНОГО КУРСА

МДК 01.01 Разработка программных модулей

1 Для текущего и рубежного контроля освоения дисциплинарных компетенций используются следующие методы:

- Устный опрос
- Наблюдение и оценка результатов практических занятий
- Наблюдение и оценка результатов лабораторных занятий
- Наблюдение и оценка результатов курсового проекта (работы)
- Экспертная оценка результатов самостоятельной работы
- Экспертная оценка по результатам наблюдения за деятельностью обучающегося в

процессе освоения ПМ

2 Формой контроля промежуточной аттестации междисциплинарного курса являются: Другая форма контроля – 4 семестр; Экзамен – 5 семестр; Дифференцированный зачет – 6 семестр; Экзамен – 7 семестр, которые проводятся в сроки, установленные учебным планом и определяемые календарным учебным графиком образовательного процесса.

Таблица 1 – Методы и формы контроля и оценивания элементов междисциплинарного курса
МДК 01.01 Разработка программных модулей

Элемент учебной дисциплины	Методы и формы контроля и оценивания		
	Текущий контроль	Рубежный контроль	Промежуточная аттестация
Раздел 1			
Тема 1.1 Жизненный цикл ПО	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	Тестирование	Другая форма контроля (4 семестр)
Тема 1.2 Структурное программирование			
Тема 1.3 Объектно-ориентированное программирование			
Раздел 2			Экзамен (5 семестр)
Тема 2.1 Паттерны проектирования	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	Тестирование	
Тема 2.2 Событийно-управляемое программирование			
Раздел 3			Дифференцированный зачет (6 семестр)
Тема 3.1 Оптимизация и рефакторинг кода	Устный опрос Наблюдение и оценка результатов практических занятий	Тестирование	
Тема3.2 Разработка			

пользовательского интерфейса	Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ		
Раздел 4			<i>Экзамен (7 семестр)</i>
Тема 4.1 Основы ADO.Net	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	<i>Тестирование Курсовая работа</i>	
Тема 4.2 Комплексная разработка программного обеспечения			
Курсовой проект (работа)	Наблюдение и оценка результатов курсового проекта (работы)	<i>Защита курсового проекта (работы)</i>	
Форма контроля			Экзамен

Текущий контроль усвоения материала

Текущий контроль усвоения материала проводится в форме устного опроса студентов по темам дисциплины.

Наблюдение и оценка результатов практических занятий

Типовые темы практических занятий приведены в РПД. Комплект заданий на практические занятия приведены в МУ по ПЗ по учебной дисциплине.

Защита отчетов по практическим занятиям проводится индивидуально каждым обучающимся в форме собеседования.

Наблюдение и оценка результатов лабораторных занятий

Типовые темы лабораторных работ приведены в РПД. Комплект заданий на лабораторные занятия приведены в МУ по ЛР по учебной дисциплине.

Защита отчетов по лабораторным занятиям проводится индивидуально каждым обучающимся в форме собеседования.

Наблюдение и оценка результатов курсового проекта (работы)

Типовые темы для разработки курсового проекта (работы) приведены в РП ПМ. Комплект заданий и указания к выполнению курсового проекта (работы) приведены в МУ по КП.

Защита курсового проекта (работы) проводится индивидуально каждым обучающимся в форме собеседования.

Экспертная оценка результатов самостоятельной работы

Задания для самостоятельной работы приведены в МУ по СРС по учебной дисциплине.

Качественная оценка определения научного кругозора, степенью овладения методами теоретического исследования и развития самостоятельности мышления студента.

Способом проверки качества организации самостоятельной работы студентов является контроль:

- корректирующий (может осуществляться во время индивидуальных консультаций по поводу выполнения формы самостоятельной работы);
- констатирующий (по результатам выполнения специальных форм самостоятельной работы);
- самоконтроль (осуществляется самим студентом);
- текущий (в ходе выполнения различных форм самостоятельной работы, установленных рабочей программой);
- промежуточный (оценка результата обучения как итога выполнения студентом всех форм самостоятельной работы).

Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины

Осуществляется как наблюдение за процессом деятельности обучающегося в режиме реального времени. Является качественной оценкой освоения учебной дисциплины, учитываемой при промежуточной аттестации.

Рубежный контроль

Рубежный контроль для комплексного оценивания усвоенных знаний, усвоенных умений проводится в форме, тестирования (после изучения разделов учебной дисциплины).

2 РЕЗУЛЬТАТЫ ОСВОЕНИЯ ЭЛЕМЕНТОВ МЕЖДИСЦИПЛИНАРНОГО КУРСА, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ ПРИ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

В результате промежуточной аттестации междисциплинарного курса осуществляется комплексная проверка следующих умений, знаний:

Результаты обучения (освоенные умения, усвоенные знания)	Показатели оценки результатов
Уметь:	
Осуществлять разработку кода программного модуля на языках низкого и высокого уровней	Умеет осуществлять разработку кода программного модуля на языках низкого и высокого уровней
Создавать программу по разработанному алгоритму как отдельный модуль	Умеет создавать программу по разработанному алгоритму как отдельный модуль
Знать:	
Основные этапы разработки программного обеспечения	Знает основные этапы разработки программного обеспечения
Основные принципы технологии структурного и объектно-ориентированного программирования	Знает основные принципы технологии структурного и объектно-ориентированного программирования

3 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ ЗНАНИЙ И УМЕНИЙ

МДК 01.01 Разработка программных модулей

Задания для оценки освоения *Раздела 1*

Тема 1.1 Жизненный цикл ПО

Обучающийся должен

знать:

- Основные этапы разработки программного обеспечения
- Основные принципы технологии структурного и объектно-ориентированного программирования.

уметь:

- Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- Создавать программу по разработанному алгоритму как отдельный модуль.

Типовые вопросы для устного опроса

1. Что такое жизненный цикл ПО?
2. Какие основные этапы включает жизненный цикл ПО?
3. Какой из этапов жизненного цикла ПО считается наиболее критическим?
4. Какие модели жизненного цикла ПО существуют?
5. Какую модель жизненного цикла ПО вы бы выбрали для проекта, требующего высокой надежности и стабильности?
6. Какие инструменты могут использоваться для управления жизненным циклом ПО?
7. Какие роли и задачи назначаются на каждом этапе жизненного цикла ПО?
8. Что такое тестирование ПО и как оно связано с жизненным циклом ПО?
9. Какие методологии разработки ПО основаны на жизненном цикле ПО?
10. Какие преимущества может принести правильное управление жизненным циклом ПО для бизнеса и пользователей?

Типовой тест

Тема 1.1 Жизненный цикл ПО

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью *Onlinetestpad.com*

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Контрольные вопросы для проведения тестирования

1. Какой этап жизненного цикла ПО отвечает за сбор требований и определение функциональных возможностей продукта?
А. Анализ требований

- B. Проектирование
 - C. Разработка
 - D. Тестирование
2. Какой этап жизненного цикла ПО отвечает за создание технического задания и дизайна продукта?
 - A. Анализ требований
 - B. Проектирование
 - C. Разработка
 - D. Тестирование
 3. Какой этап жизненного цикла ПО отвечает за написание кода?
 - A. Анализ требований
 - B. Проектирование
 - C. Разработка
 - D. Тестирование
 4. Какой этап жизненного цикла ПО отвечает за проверку и исправление ошибок?
 - A. Анализ требований
 - B. Проектирование
 - C. Разработка
 - D. Тестирование
 5. Какой этап жизненного цикла ПО отвечает за тестирование и проверку работоспособности продукта?
 - A. Анализ требований
 - B. Проектирование
 - C. Разработка
 - D. Тестирование
 6. Какой этап жизненного цикла ПО отвечает за установку и настройку продукта на компьютерах пользователей?
 - A. Анализ требований
 - B. Проектирование
 - C. Разработка
 - D. Внедрение
 7. Какой этап жизненного цикла ПО отвечает за обучение пользователей работе с продуктом?
 - A. Анализ требований
 - B. Проектирование
 - C. Разработка
 - D. Внедрение
 8. Какой этап жизненного цикла ПО отвечает за поддержку и сопровождение продукта после его выпуска?
 - A. Анализ требований
 - B. Проектирование
 - C. Разработка
 - D. Эксплуатация
 9. Какой этап жизненного цикла ПО отвечает за проведение и анализ результатов опросов пользователей?
 - A. Анализ требований

- В. Проектирование
 - С. Разработка
 - Д. Эксплуатация
10. Какой этап жизненного цикла ПО отвечает за принятие решения о дальнейшей судьбе продукта?
- А. Анализ требований
 - В. Проектирование
 - С. Разработка
 - Д. Эксплуатация

Тема 1.2 Структурное программирование

Обучающийся должен

знать:

- Основные этапы разработки программного обеспечения
- Основные принципы технологии структурного и объектно-ориентированного программирования.

уметь:

- Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- Создавать программу по разработанному алгоритму как отдельный модуль.

Типовые вопросы для устного опроса

1. Что такое структурное программирование?
2. Какие основные принципы лежат в основе структурного программирования?
3. Какие языки программирования поддерживают структурное программирование?
4. Какие структуры данных используются в структурном программировании?
5. Какие преимущества может принести использование структурного программирования в разработке ПО?
6. Какие недостатки может иметь структурное программирование?
7. Какие методы тестирования ПО могут использоваться при структурном программировании?
8. Какие инструменты могут помочь в разработке программного кода в стиле структурного программирования?
9. Какие роли и задачи могут быть назначены в команде, работающей в стиле структурного программирования?
10. Какие современные методологии разработки ПО включают принципы структурного программирования?

Типовой тест

Тема 1.2 Структурное программирование

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью [Onlinetestpad.com](https://onlinetestpad.com)

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Контрольные вопросы для проведения тестирования

1. Что такое структурное программирование?
 - A) Подход к разработке программ, основанный на использовании только процедур и функций
 - B) Подход к разработке программ, основанный на использовании объектов и классов
 - C) Подход к разработке программ, основанный на использовании глобальных переменных
 - D) Подход к разработке программ, основанный на использовании условных операторов
2. Какие основные конструкции используются в структурном программировании?
 - A) Функции и классы
 - B) Циклы и условные операторы
 - C) Глобальные переменные и объекты
 - D) Интерфейсы и абстрактные классы
3. Какие преимущества имеет структурное программирование?
 - A) Простота и понятность кода
 - B) Возможность повторного использования кода
 - C) Большая скорость выполнения программы
 - D) Возможность создания сложных иерархий наследования
4. Какие недостатки имеет структурное программирование?
 - A) Ограниченность в создании сложных структур данных
 - B) Трудность в отладке больших программных проектов
 - C) Сложность в создании кроссплатформенных программ
 - D) Отсутствие поддержки многопоточности
5. Какое ключевое слово используется для определения функции в языке программирования C?
 - A) def
 - B) function
 - C) void
 - D) return
6. Какое ключевое слово используется для объявления переменной в языке программирования Java?
 - A) int
 - B) var
 - C) string
 - D) double
7. Какое ключевое слово используется для описания цикла for в языке программирования Python?
 - A) for
 - B) repeat
 - C) until
 - D) while
8. Какая структура данных используется для хранения элементов в порядке FIFO (First-In-First-Out)?

- A) Стек
 - B) Очередь
 - C) Список
 - D) Массив
9. Какое ключевое слово используется для объявления структуры в языке программирования C++?
- A) structure
 - B) class
 - C) record
 - D) type
10. Какое ключевое слово используется для описания условного оператора в языке программирования JavaScript?
- A) if
 - B) switch
 - C) else
 - D) case

Тема 1.3 Объектно-ориентированное программирование

Обучающийся должен

знать:

- Основные этапы разработки программного обеспечения
- Основные принципы технологии структурного и объектно-ориентированного программирования.

уметь:

- Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- Создавать программу по разработанному алгоритму как отдельный модуль.

Типовые вопросы для устного опроса

1. Что такое объектно-ориентированное программирование (ООП)?
2. Какие основные принципы лежат в основе ООП?
3. Какие языки программирования поддерживают ООП?
4. Какие структуры данных используются в ООП?
5. Какие преимущества может принести использование ООП в разработке ПО?
6. Какие недостатки может иметь ООП?
7. Какие методы тестирования ПО могут использоваться при ООП?
8. Какие инструменты могут помочь в разработке программного кода в стиле ООП?
9. Какие роли и задачи могут быть назначены в команде, работающей в стиле ООП?
10. Какие современные методологии разработки ПО включают принципы ООП?

Типовой тест

Тема 1.3 Объектно-ориентированное программирование

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью [Onlinetestpad.com](https://onlinetestpad.com)

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Контрольные вопросы для проведения тестирования

1. Что такое объектно-ориентированное программирование (ООП)?
 - A) Подход к разработке программ, основанный на использовании только процедур и функций
 - B) Подход к разработке программ, основанный на использовании объектов и классов
 - C) Подход к разработке программ, основанный на использовании глобальных переменных
 - D) Подход к разработке программ, основанный на использовании условных операторов
2. Какие основные концепции используются в объектно-ориентированном программировании?
 - A) Процедуры и функции
 - B) Объекты и классы
 - C) Глобальные переменные и объекты
 - D) Интерфейсы и абстрактные классы
3. Какие преимущества имеет объектно-ориентированное программирование?
 - A) Простота и понятность кода
 - B) Возможность повторного использования кода
 - C) Большая скорость выполнения программы
 - D) Возможность создания сложных иерархий наследования
4. Какие недостатки имеет объектно-ориентированное программирование?
 - A) Ограниченность в создании сложных структур данных
 - B) Трудность в отладке больших программных проектов
 - C) Сложность в создании кроссплатформенных программ
 - D) Отсутствие поддержки многопоточности
5. Какое ключевое слово используется для создания нового объекта в языке программирования Java?
 - A) new
 - B) create
 - C) object
 - D) instance
6. Какой модификатор доступа позволяет обращаться к членам класса из любой точки программы?
 - A) private
 - B) protected
 - C) public
 - D) static
7. Какое ключевое слово используется для наследования класса в языке программирования C#?
 - A) extends
 - B) inherits
 - C) implements
 - D) interfaces

8. Какая концепция ООП описывает возможность класса принимать форму любого наследника?
 - A) Инкапсуляция
 - B) Наследование
 - C) Полиморфизм
 - D) Абстракция
9. Какой метод вызывается автоматически при создании нового объекта класса в языке программирования Python?
 - A) init()
 - B) constructor()
 - C) create()
 - D) object()
10. Какой модификатор доступа позволяет обращаться к членам класса только изнутри этого класса?
 - A) private
 - B) protected
 - C) public
 - D) static

Задания для оценки освоения Раздела 2

Тема 2.1 Паттерны проектирования

Обучающийся должен

знать:

- Основные этапы разработки программного обеспечения
- Основные принципы технологии структурного и объектно-ориентированного программирования.

уметь:

- Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- Создавать программу по разработанному алгоритму как отдельный модуль.

Типовые вопросы для устного опроса

1. Что такое паттерны проектирования?
2. Какие основные категории паттернов проектирования существуют?
3. Какие преимущества может принести использование паттернов проектирования в разработке ПО?
4. Какие языки программирования поддерживают паттерны проектирования?
5. Какие паттерны проектирования относятся к созданию объектов?
6. Какие паттерны проектирования относятся к структурированию классов?
7. Какие паттерны проектирования относятся к управлению поведением объектов?
8. Какие паттерны проектирования относятся к взаимодействию объектов?
9. Какие роли и задачи могут быть назначены в команде, работающей с паттернами проектирования?
10. Какие современные методологии разработки ПО включают использование паттернов проектирования?

11. Что такое событийно-управляемое программирование (event-driven programming)?
12. Какие компоненты могут генерировать события в программе?
13. Какие обработчики событий могут быть присоединены к компонентам в программе?
14. Как можно отличить событие от обычного метода вызова в программе?
15. Какие преимущества имеет событийно-управляемое программирование по сравнению с процедурным подходом?
16. Какие недостатки могут быть у событийно-управляемого программирования?
17. Какие технологии и языки программирования поддерживают событийно-управляемое программирование?
18. Какие особенности разработки пользовательского интерфейса связаны с применением событийно-управляемого подхода?
19. Какие типы событий могут возникать в web-приложениях?
20. Как можно организовать обработку нескольких событий одновременно в программе?

Типовой тест

Тема 2.1 Паттерны проектирования

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью *Onlinetestpad.com*

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Контрольные вопросы для проведения тестирования

1. Что такое паттерны проектирования?
 - A) Модели проектирования баз данных
 - B) Шаблоны для проектирования архитектуры программного обеспечения
 - C) Инструменты для тестирования программного обеспечения
 - D) Алгоритмы оптимизации производительности
2. Какие основные категории паттернов проектирования существуют?
 - A) Паттерны архитектуры, структурные паттерны и паттерны поведения
 - B) Паттерны реализации, тестирования и оптимизации
 - C) Паттерны взаимодействия с базами данных, сетевыми протоколами и операционной системой
 - D) Паттерны кодирования, отладки и документирования
3. Какой паттерн проектирования используется для создания объектов с определенными свойствами и методами?
 - A) Singleton
 - B) Prototype
 - C) Builder
 - D) Decorator
4. Какой паттерн проектирования используется для разделения функциональности между объектами, снижения связности и повышения переиспользуемости кода?
 - A) Observer

- B) Composite
 - C) Mediator
 - D) Adapter
5. Какой паттерн проектирования используется для управления изменяемым состоянием объекта и уведомлении других объектов об изменениях?
- A) Observer
 - B) State
 - C) Strategy
 - D) Decorator
6. Какой паттерн проектирования используется для организации объектов в иерархические структуры и упрощения работы с ними?
- A) Builder
 - B) Decorator
 - C) Iterator
 - D) Composite
7. Какой паттерн проектирования используется для создания новых объектов на основе существующих, без явного использования ключевого слова new?
- A) Factory Method
 - B) Prototype
 - C) Abstract Factory
 - D) Builder
8. Какой паттерн проектирования используется для определения семейства взаимосвязанных алгоритмов и выбора из них на лету?
- A) Observer
 - B) Strategy
 - C) Template Method
 - D) Visitor
9. Какой паттерн проектирования используется для изменения функциональности объекта без изменения его основного класса?
- A) Decorator
 - B) Proxy
 - C) Adapter
 - D) Bridge
10. Какой паттерн проектирования используется для уменьшения количества объектов, создаваемых в приложении, и обеспечения доступа к одному экземпляру класса?
- A) Singleton
 - B) Facade
 - C) Mediator
 - D) Flyweight

Тема 2.2 Событийно-управляемое программирование

Обучающийся должен

знать:

- Основные этапы разработки программного обеспечения
- Основные принципы технологии структурного и объектно-ориентированного программирования.

уметь:

- Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- Создавать программу по разработанному алгоритму как отдельный модуль.

Типовые вопросы для устного опроса

1. Что такое событийно-управляемое программирование?
2. Какие основные принципы лежат в основе событийно-управляемого программирования?
3. Какие языки программирования поддерживают событийно-управляемое программирование?
4. Какие объекты могут быть источниками событий в событийно-управляемом программировании?
5. Какие преимущества может принести использование событийно-управляемого программирования в разработке ПО?
6. Какие недостатки может иметь событийно-управляемое программирование?
7. Какие методы тестирования ПО могут использоваться при событийно-управляемом программировании?
8. Какие инструменты могут помочь в разработке программного кода в стиле событийно-управляемого программирования?
9. Какие роли и задачи могут быть назначены в команде, работающей в стиле событийно-управляемого программирования?
10. Какие современные методологии разработки ПО включают принципы событийно-управляемого программирования?

Типовой тест

Тема 2.2 Событийно-управляемое программирование

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью [Onlinetestpad.com](https://onlinetestpad.com)

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Контрольные вопросы для проведения тестирования

1. Что такое событийно-управляемое программирование (Event-driven programming)?
 - A) Методика написания кода, использующая только события
 - B) Способ написания асинхронного кода
 - C) Программирование, основанное на взаимодействии объектов через события
 - D) Программирование, использующее только один поток выполнения
2. Каким образом работает событийная модель в JavaScript?
 - A) События возникают только при выполнении определенных действий пользователей
 - B) События могут возникать как при взаимодействии пользователя с элементами на странице, так и при выполнении определенных действий в коде
 - C) События возникают только при выполнении определенных действий в коде

- D) События могут возникать только при взаимодействии пользователя с элементами на странице
3. Какие преимущества имеет событийно-управляемое программирование?
- A) Более простой и понятный код
 - B) Увеличение производительности приложения
 - C) Более быстрый запуск приложения
 - D) Увеличение устойчивости приложения к ошибкам
4. Какой объект используется для регистрации событий в C#?
- A) Delegate
 - B) Event
 - C) Handler
 - D) Action
5. Какой метод в Python используется для регистрации обработчиков событий?
- A) on()
 - B) add_listener()
 - C) bind()
 - D) connect()
6. Какой тип события используется для отслеживания изменения состояния элемента на странице?
- A) mouseover
 - B) click
 - C) keypress
 - D) onchange
7. Какой тип события используется для отслеживания изменения значения input элемента на странице?
- A) mouseover
 - B) click
 - C) keypress
 - D) onchange
8. Какой метод в JavaScript используется для удаления обработчика событий?
- A) removeEventListener()
 - B) deleteListener()
 - C) unbind()
 - D) detach()
9. Какой паттерн проектирования используется для управления событиями в приложении?
- A) Observer
 - B) Singleton
 - C) Factory Method
 - D) Prototype
10. Каким образом события влияют на выполнение программы?
- A) События блокируют выполнение программы
 - B) События останавливают выполнение программы на неопределенный период времени
 - C) События позволяют выполнить необходимый код только при наступлении определенных условий
 - D) События не влияют на выполнение программы

Задания для оценки освоения Раздела 3

Тема 3.1 Оптимизация и рефакторинг кода

Обучающийся должен

знать:

- Основные этапы разработки программного обеспечения
- Основные принципы технологии структурного и объектно-ориентированного программирования.

уметь:

- Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- Создавать программу по разработанному алгоритму как отдельный модуль.

Типовые вопросы для устного опроса

1. Что такое оптимизация кода?
2. Какие методы оптимизации кода существуют?
3. Какие преимущества может принести оптимизация кода в разработке ПО?
4. Какие риски может иметь оптимизация кода?
5. Что такое рефакторинг кода?
6. Какие методы рефакторинга кода существуют?
7. Какие преимущества может принести рефакторинг кода в разработке ПО?
8. Какие риски может иметь рефакторинг кода?
9. Какие инструменты могут помочь в оптимизации и рефакторинге кода?
10. Какие роли и задачи могут быть назначены в команде, работающей над оптимизацией и рефакторингом кода?
11. Какую роль играет пользовательский интерфейс в разработке программного обеспечения?
12. Какие принципы лежат в основе разработки удобного пользовательского интерфейса?
13. Какие методы и инструменты используются при проектировании пользовательского интерфейса?
14. Какие тренды в современном дизайне пользовательского интерфейса вы можете выделить?
15. Каким образом дизайнер учитывает психологию пользователя при создании пользовательского интерфейса?
16. Какие проблемы могут возникать при разработке мультязычного пользовательского интерфейса?
17. Какие методы тестирования пользовательского интерфейса существуют?
18. Как можно оценить качество пользовательского интерфейса?
19. Какова роль пользовательского интерфейса в создании мобильных приложений?
20. Какие принципы разработки пользовательского интерфейса могут быть использованы для создания интерфейса голосовых помощников?

Типовой тест

Тема 3.1 Оптимизация и рефакторинг кода

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью [Onlinetestpad.com](https://onlinetestpad.com)

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Контрольные вопросы для проведения тестирования

1. Что такое оптимизация кода?
 - A) Изменение кода с целью улучшения его читаемости
 - B) Изменение кода с целью ускорения его выполнения
 - C) Изменение кода с целью улучшения его стиля
 - D) Изменение кода с целью добавления новой функциональности
2. Какой подход к оптимизации кода рекомендуется использовать в первую очередь?
 - A) Использование быстрых алгоритмов
 - B) Удаление дублирующегося кода
 - C) Использование эффективных структур данных
 - D) Избегание излишней оптимизации
3. Что такое рефакторинг кода?
 - A) Изменение кода с целью добавления новой функциональности
 - B) Изменение кода с целью улучшения его стиля
 - C) Изменение кода с целью улучшения его читаемости и поддерживаемости без изменения функциональности
 - D) Изменение кода с целью ускорения его выполнения
4. Какие инструменты можно использовать для оптимизации и рефакторинга кода?
 - A) IDE и отладчики
 - B) Библиотеки и фреймворки
 - C) Компиляторы и интерпретаторы
 - D) Тестовые фреймворки и библиотеки
5. Какой паттерн проектирования можно использовать для устранения дублирующегося кода?
 - A) Singleton
 - B) Factory Method
 - C) Prototype
 - D) Template Method
6. Какой инструмент можно использовать для измерения производительности кода?
 - A) IDE
 - B) Отладчик
 - C) Профилировщик
 - D) Компилятор
7. Что такое "бутылочное горлышко" (bottleneck) в программном коде?
 - A) Участок кода, который может быть оптимизирован без потери функциональности
 - B) Участок кода, который является узким местом при выполнении программы
 - C) Участок кода, который необходимо переписать для улучшения его читаемости
 - D) Участок кода, который не используется в программе
8. Какие проблемы могут возникнуть при излишней оптимизации кода?
 - A) Увеличение производительности программы
 - B) Увеличение размера и сложности кода

- C) Снижение устойчивости программы к ошибкам
 - D) Улучшение читаемости и понятности кода
9. Какие преимущества может дать рефакторинг кода?
- A) Улучшение производительности программы
 - B) Улучшение читаемости и поддерживаемости кода
 - C) Добавление новой функциональности
 - D) Улучшение дизайна пользовательского интерфейса
10. Какие техники оптимизации кода могут использоваться для уменьшения времени выполнения программы?
- A) Использование кэшей и буферов
 - B) Оптимизация графических ресурсов
 - C) Удаление ненужных файлов и данных
 - D) Использование улучшенных алгоритмов и структур данных

Тема 3.2 Разработка пользовательского интерфейса

Обучающийся должен

знать:

- Основные этапы разработки программного обеспечения
- Основные принципы технологии структурного и объектно-ориентированного программирования.

уметь:

- Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- Создавать программу по разработанному алгоритму как отдельный модуль.

Типовые вопросы для устного опроса

1. Что такое пользовательский интерфейс?
2. Какие виды пользовательского интерфейса существуют?
3. Какие принципы следует учитывать при разработке пользовательского интерфейса?
4. Какие инструменты могут помочь в разработке пользовательского интерфейса?
5. Какие преимущества может принести хороший пользовательский интерфейс для разработки ПО?
6. Какие недостатки может иметь плохой пользовательский интерфейс?
7. Какие методы тестирования пользовательского интерфейса могут использоваться?
8. Какие требования должны быть учтены при разработке пользовательского интерфейса для мобильных устройств?
9. Какие роли и задачи могут быть назначены в команде, работающей над разработкой пользовательского интерфейса?
10. Какие современные тренды в разработке пользовательского интерфейса существуют?

Типовой тест

Тема 3.2 Разработка пользовательского интерфейса

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью [Onlinetestpad.com](https://onlinetestpad.com)

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Контрольные вопросы для проведения тестирования

1. Что такое пользовательский интерфейс?
 - A) Набор инструментов для разработки программного обеспечения
 - B) Интерфейс между компьютером и пользователями
 - C) База данных, хранящая информацию о пользователях
 - D) Система безопасности для пользователей
2. Какие виды пользовательских интерфейсов существуют?
 - A) Графический и текстовый
 - B) Командный и файловый
 - C) Аналоговый и цифровой
 - D) Сетевой и локальный
3. Что такое прототипирование в разработке пользовательского интерфейса?
 - A) Создание полной версии интерфейса до начала программирования
 - B) Создание базового макета интерфейса для оценки функциональности и дизайна
 - C) Удаление неиспользуемых элементов интерфейса
 - D) Изменение цветовой гаммы интерфейса в соответствии с требованиями клиента
4. Что такое макет пользовательского интерфейса?
 - A) Описание дизайна и структуры интерфейса
 - B) Таблица, содержащая информацию обо всех элементах интерфейса
 - C) Список требований к функциональности интерфейса
 - D) Код, описывающий работу интерфейса
5. Какие основные элементы пользовательского интерфейса существуют?
 - A) Кнопки, поля ввода, выпадающие списки
 - B) Базы данных, серверы, клиенты
 - C) Графики, звуки, анимация
 - D) Чаты, форумы, блоги
6. Что такое доступность интерфейса?
 - A) Возможность интерфейса работать с разными операционными системами
 - B) Возможность использования интерфейса людьми с различными ограничениями
 - C) Возможность открытия и редактирования исходного кода интерфейса
 - D) Возможность удаленного доступа к интерфейсу
7. Какие технологии используются для создания графического пользовательского интерфейса?
 - A) HTML и CSS
 - B) Python и Ruby
 - C) JavaScript и PHP
 - D) SQL и NoSQL
8. Что такое адаптивный дизайн интерфейса?
 - A) Дизайн интерфейса, который изменяется в зависимости от времени суток
 - B) Дизайн интерфейса, который меняется в зависимости от местоположения пользователя
 - C) Дизайн интерфейса, который адаптируется к разным
9. Что такое UX-дизайн?

- А) Дизайн, ориентированный на создание приятного визуального образа
 - В) Дизайн, ориентированный на удобство использования и удовлетворение потребностей пользователя
 - С) Дизайн, ориентированный на создание максимально функциональных интерфейсов
 - Д) Дизайн, ориентированный на создание эффективных и экономичных систем
10. Что такое контекстная помощь в пользовательском интерфейсе?
- А) Справочная система, предоставляющая информацию о том, как использовать элементы интерфейса
 - В) Кнопка, которая помогает пользователям поделиться приложением в социальных сетях
 - С) Элемент интерфейса, который позволяет пользователю сменить язык интерфейса
 - Д) Всплывающее окно с предупреждением, что пользователь может потерять несохраненные данные при закрытии программы

Задания для оценки освоения Раздела 4

Тема 4.1 Основы ADO.Net

Обучающийся должен

знать:

- Основные этапы разработки программного обеспечения
- Основные принципы технологии структурного и объектно-ориентированного программирования.

уметь:

- Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- Создавать программу по разработанному алгоритму как отдельный модуль.

Типовые вопросы для устного опроса

1. Что такое ADO.Net и для чего он используется?
2. Какие преимущества предоставляет ADO.Net в работе с базами данных?
3. Какие объекты ADO.Net используются для подключения к базе данных?
4. Какие объекты ADO.Net используются для выполнения запросов к базе данных?
5. Какие типы команд для работы с базой данных можно использовать в ADO.Net?
6. Что такое объектный тип доступа к данным в ADO.Net?
7. Какие технологии могут использоваться вместе с ADO.Net для работы с базой данных?
8. Какие роли и задачи могут быть назначены в команде, работающей с ADO.Net?
9. Какие методы тестирования ADO.Net могут использоваться?
10. Какие современные тренды в работе с базами данных с помощью ADO.Net существуют?
11. Что такое комплексная разработка программного обеспечения?
12. Какие этапы включает в себя комплексная разработка программного обеспечения?
13. Какие роли в команде разработки могут присутствовать при комплексной разработке ПО?
14. Какие основные методологии разработки программного обеспечения могут использоваться при комплексной разработке ПО?
15. Какие инструменты используются для управления процессом разработки ПО при комплексной разработке?
16. Какие методы тестирования используются при комплексной разработке ПО?
17. Каковы основные вызовы, связанные с комплексной разработкой ПО?

18. Какие технологии и языки программирования могут использоваться при комплексной разработке ПО?
19. Какие задачи могут решаться с помощью комплексной разработки ПО?
20. Каковы факторы, которые могут повлиять на успех комплексной разработки ПО?

Типовой тест

Тема 4.1 Основы ADO.Net

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью Onlinetestpad.com

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Контрольные вопросы для проведения тестирования

1. Что означает аббревиатура ADO.Net?
 - A) ActiveX Data Objects.Net
 - B) Advanced Data Objects.Net
 - C) Active Data Objects.Net
 - D) Abstract Data Objects.Net
2. Какие компоненты входят в ADO.Net?
 - A) OleDb, Odbc, DataSet, DataTable
 - B) SqlConnection, SqlCommand, SqlDataReader, SqlDataAdapter
 - C) LINQ, Entity Framework, ADO, XML
 - D) DataList, DataView, DataGrid, DataSource
3. Какие объекты используются для подключения к базе данных в ADO.Net?
 - A) OleDbConnection, SqlConnection, OdbcConnection
 - B) SqlConnection, SqlCommand, SqlDataReader
 - C) DataContext, EntityConnection,ObjectContext
 - D) XmlDocument, XmlDataSource, XmlReader
4. Какие объекты используются для выполнения SQL-запросов в ADO.Net?
 - A) OleDbCommand, SqlCommand, OdbcCommand
 - B) SqlCommand, SqlDataReader, SqlDataAdapter
 - C) LinqCommand, EntityCommand, ObjectCommand
 - D) XmlCommand, XmlTextReader, XmlTextWriter
5. Что такое DataSet в ADO.Net?
 - A) Объект, представляющий таблицу в памяти приложения
 - B) Объект, представляющий базу данных в памяти приложения
 - C) Объект, представляющий результат SQL-запроса в памяти приложения
 - D) Объект, представляющий набор таблиц в памяти приложения
6. Что такое DataAdapter в ADO.Net?
 - A) Объект, используемый для выполнения SQL-запросов и получения данных из базы данных
 - B) Объект, используемый для выполнения CRUD-операций над таблицей

- C) Объект, используемый для управления транзакциями в базе данных
D) Объект, используемый для валидации данных перед записью в базу данных
7. Что такое SqlCommandBuilder в ADO.Net?
A) Объект, используемый для автоматической генерации SQL-запросов на основе изменений в DataSet
B) Объект, используемый для создания динамических SQL-запросов в коде приложения
C) Объект, используемый для создания хранимых процедур в базе данных
D) Объект, используемый для выполнения SQL-запросов и получения данных из базы данных
8. Какие типы соединений поддерживаются в ADO.Net?
A) Inner join, Left join, Right join, Full join
B) Primary key join, Foreign key join, Self join, Cross join
C) SqlConnection, OleDbConnection, OdbcConnection
D) One-to-one, One-to-many, Many-to-many, Many-to-one
9. Что такое хеш-таблица?
a) Структура данных, позволяющая хранить и обрабатывать данные в виде пар "ключ-значение"
b) Массив данных, упорядоченный по значению ключа
c) Структура данных, используемая для хранения деревьев
d) Список элементов, упорядоченный по значению ключа
10. Какой алгоритм сортировки наиболее эффективен для больших массивов данных?
a) Пузырьковая сортировка
b) Сортировка выбором
c) Быстрая сортировка
d) Сортировка вставками

Тема 4.2 Комплексная разработка программного обеспечения

Обучающийся должен

знать:

— Основные этапы разработки программного обеспечения

— Основные принципы технологии структурного и объектно-ориентированного программирования.

уметь:

— Осуществлять разработку кода программного модуля на языках низкого и высокого уровней;

— Создавать программу по разработанному алгоритму как отдельный модуль.

Типовые вопросы для устного опроса

1. Что такое комплексная разработка программного обеспечения?
2. Какие этапы включает в себя комплексная разработка программного обеспечения?
3. Какие методологии разработки ПО могут использоваться в комплексной разработке программного обеспечения?
4. Какие роли и задачи могут быть назначены в команде, работающей над комплексной разработкой программного обеспечения?
5. Какие инструменты могут использоваться в комплексной разработке программного обеспечения?

6. Какие принципы следует учитывать при выборе технологий для комплексной разработки программного обеспечения?
7. Какие методы тестирования могут использоваться в комплексной разработке программного обеспечения?
8. Какие преимущества может принести комплексная разработка программного обеспечения для заказчика?
9. Какие риски могут быть связаны с комплексной разработкой программного обеспечения?
10. Какие современные тренды в комплексной разработке программного обеспечения существуют?

Типовой тест

Тема 4.2 Комплексная разработка программного обеспечения

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью *Onlinetestpad.com*

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Контрольные вопросы для проведения тестирования

1. Какие из следующих моделей жизненного цикла наиболее подходят для комплексной разработки программного обеспечения?
 - a) Модель каскада
 - b) Итеративная модель
 - c) Agile-модель
 - d) Только a) и b)
2. Что такое Continuous Integration?
 - a) Автоматическое тестирование кода при каждом изменении
 - b) Автоматическое развертывание новых версий приложения
 - c) Автоматическое исправление ошибок в коде
 - d) Автоматическое создание документации по проекту
3. Какие инструменты используются для управления проектами в комплексной разработке ПО?
 - a) Git
 - b) Jira
 - c) Visual Studio
 - d) Только a) и b)
4. Что такое code review?
 - a) Процесс, при котором другие разработчики проверяют ваш код на ошибки и проблемы
 - b) Процесс, при котором вы проверяете свой код на ошибки и проблемы
 - c) Процесс, при котором код тестируется на соответствие спецификациям
 - d) Процесс, при котором код проверяется на соответствие стилю написания
5. Какие этапы включает процесс комплексной разработки программного обеспечения?
 - a) Анализ и проектирование
 - b) Реализация

- c) Тестирование и отладка
 - d) Все вышеперечисленные
6. Что такое UML?
- a) Язык моделирования
 - b) Язык программирования
 - c) Язык разметки
 - d) Язык запросов к базам данных
7. Что такое DevOps?
- a) Философия, объединяющая разработку и операционную деятельность
 - b) Методология разработки программного обеспечения
 - c) Инструментарий для автоматического тестирования ПО
 - d) Фреймворк для разработки веб-приложений
8. Какие преимущества имеет подход DevOps в комплексной разработке программного обеспечения?
- a) Уменьшение времени между сдачей новой функциональности и ее выкаткой в продакшен
 - b) Более частое тестирование нового кода и более быстрое выявление проблем
 - c) Улучшение коммуникации между разработчиками и операционной командой
 - d) Все вышеперечисленные
9. Что такое контроль версий и какой инструмент используется для этой цели в комплексной разработке программного обеспечения?
- a) Контроль версий - это способ управления изменениями в исходном коде, используется инструмент Git
 - b) Контроль версий - это способ отслеживания работы разработчиков, используется инструмент Jira
 - c) Контроль версий - это способ тестирования кода, используется инструмент Selenium
 - d) Контроль версий - это способ управления доступом к коду, используется инструмент Visual Studio
10. Какие этапы включает Agile-модель в комплексной разработке программного обеспечения?
- a) Планирование и анализ
 - b) Разработка
 - c) Тестирование
 - d) Все вышеперечисленные

4 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

МДК 01.01 Разработка программных модулей

Изучение ***МДК 01.01 Разработка программных модулей*** реализуется в течение 4 семестров.

Формой контроля промежуточной аттестации ***МДК 01.01 Разработка программных модулей*** являются другая форма контроля – 4 семестр, экзамены – 5,7 семестры дифференцированный зачет – 6 семестр.

Основой для определения оценки при проведении промежуточных аттестаций служит объём и уровень усвоения обучающимися материала, предусмотренного рабочей программой профессионального модуля ***ПМ.01 Разработка модулей программного обеспечения для компьютерных систем*** в части ***МДК 01.01 Разработка программных модулей***.

Другая форма контроля (4 семестр)

Другая форма контроля проводится по завершению изучения раздела МДК по окончании семестра защиты отчетов по лабораторным работам

Экзамен (5 семестр)

Экзамен проводится по завершению изучения раздела МДК по окончании семестра в форме собеседования.

К сдаче экзамена допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации за семестр оценки не ниже «удовлетворительно».

Вопросы для подготовки к экзамену

Перечень вопросов для оценки усвоенных знаний

1. Что такое паттерны проектирования?
2. Какие основные категории паттернов проектирования существуют?
3. Какие преимущества может принести использование паттернов проектирования в разработке ПО?
4. Какие языки программирования поддерживают паттерны проектирования?
5. Какие паттерны проектирования относятся к созданию объектов?
6. Какие паттерны проектирования относятся к структурированию классов?
7. Какие паттерны проектирования относятся к управлению поведением объектов?
8. Какие паттерны проектирования относятся к взаимодействию объектов?
9. Какие роли и задачи могут быть назначены в команде, работающей с паттернами проектирования?
10. Какие современные методологии разработки ПО включают использование паттернов проектирования?
11. Что такое событийно-управляемое программирование (event-driven programming)?
12. Какие компоненты могут генерировать события в программе?
13. Какие обработчики событий могут быть присоединены к компонентам в программе?

14. Как можно отличить событие от обычного метода вызова в программе?
15. Какие преимущества имеет событийно-управляемое программирование по сравнению с процедурным подходом?
16. Какие недостатки могут быть у событийно-управляемого программирования?
17. Какие технологии и языки программирования поддерживают событийно-управляемое программирование?
18. Какие особенности разработки пользовательского интерфейса связаны с применением событийно-управляемого подхода?
19. Какие типы событий могут возникать в web-приложениях?
20. Как можно организовать обработку нескольких событий одновременно в программе?

Перечень заданий для оценки усвоенных умений

1. Напишите программу на языке программирования, которая демонстрирует использование шаблона проектирования "Одиночка" (Singleton).
2. Реализуйте фабричный метод (Factory Method) для создания объектов разных классов, объявленных в интерфейсе.
3. Напишите программу, используя шаблон проектирования "Адаптер" (Adapter), чтобы адаптировать интерфейс одного класса к другому.
4. Разработайте программу с использованием шаблона проектирования "Стратегия" (Strategy), которая демонстрирует возможность изменения поведения приложения в зависимости от выбранной стратегии.
5. Напишите программу, используя шаблон проектирования "Декоратор" (Decorator), которая позволяет динамически добавлять новые свойства или методы к объекту.
6. Напишите программу на языке программирования, которая использует события для реагирования на действия пользователя в окне приложения (например, щелчки мыши или нажатия клавиш).
7. Разработайте программу, используя событийно-ориентированное программирование, для управления потоками в приложении (например, события, которые происходят при завершении или приостановке потока).
8. Напишите программу, используя события для реагирования на изменения в базе данных, и обработки ошибок, которые могут возникнуть в процессе обработки данных.
9. Реализуйте программу, которая использует события для управления анимацией объектов в графическом интерфейсе пользователя (например, события, которые происходят при движении объекта или при завершении анимации).
10. Разработайте программу, которая использует события для реагирования на изменения в файлах или папках на диске (например, события, которые происходят при создании, изменении или удалении файла или папки).

Дифференцированный зачет (6 семестр)

Дифференцированный зачет проводится по завершению изучения раздела МДК по окончании семестра в форме теста.

К сдаче дифференцированного зачета допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации за семестр оценки не ниже «удовлетворительно».

Вопросы для подготовки к дифференцированному зачету

Перечень вопросов (тестовые задания) для оценки усвоенных знаний

1. Что такое оптимизация кода?
 - A) Изменение кода с целью улучшения его читаемости
 - B) Изменение кода с целью ускорения его выполнения
 - C) Изменение кода с целью улучшения его стиля
 - D) Изменение кода с целью добавления новой функциональности
2. Какой подход к оптимизации кода рекомендуется использовать в первую очередь?
 - A) Использование быстрых алгоритмов
 - B) Удаление дублирующегося кода
 - C) Использование эффективных структур данных
 - D) Избегание излишней оптимизации
3. Что такое рефакторинг кода?
 - A) Изменение кода с целью добавления новой функциональности
 - B) Изменение кода с целью улучшения его стиля
 - C) Изменение кода с целью улучшения его читаемости и поддерживаемости без изменения функциональности
 - D) Изменение кода с целью ускорения его выполнения
4. Какие инструменты можно использовать для оптимизации и рефакторинга кода?
 - A) IDE и отладчики
 - B) Библиотеки и фреймворки
 - C) Компиляторы и интерпретаторы
 - D) Тестовые фреймворки и библиотеки
5. Какой паттерн проектирования можно использовать для устранения дублирующегося кода?
 - A) Singleton
 - B) Factory Method
 - C) Prototype
 - D) Template Method
6. Какой инструмент можно использовать для измерения производительности кода?
 - A) IDE
 - B) Отладчик
 - C) Профилировщик
 - D) Компилятор
7. Что такое "бутылочное горлышко" (bottleneck) в программном коде?
 - A) Участок кода, который может быть оптимизирован без потери функциональности
 - B) Участок кода, который является узким местом при выполнении программы
 - C) Участок кода, который необходимо переписать для улучшения его читаемости
 - D) Участок кода, который не используется в программе
8. Какие проблемы могут возникнуть при излишней оптимизации кода?
 - A) Увеличение производительности программы

- B) Увеличение размера и сложности кода
 - C) Снижение устойчивости программы к ошибкам
 - D) Улучшение читаемости и понятности кода
9. Какие преимущества может дать рефакторинг кода?
- A) Улучшение производительности программы
 - B) Улучшение читаемости и поддерживаемости кода
 - C) Добавление новой функциональности
 - D) Улучшение дизайна пользовательского интерфейса
10. Какие техники оптимизации кода могут использоваться для уменьшения времени выполнения программы?
- A) Использование кэшей и буферов
 - B) Оптимизация графических ресурсов
 - C) Удаление ненужных файлов и данных
 - D) Использование улучшенных алгоритмов и структур данных
11. Что такое пользовательский интерфейс?
- A) Набор инструментов для разработки программного обеспечения
 - B) Интерфейс между компьютером и пользователями
 - C) База данных, хранящая информацию о пользователях
 - D) Система безопасности для пользователей
12. Какие виды пользовательских интерфейсов существуют?
- A) Графический и текстовый
 - B) Командный и файловый
 - C) Аналоговый и цифровой
 - D) Сетевой и локальный
13. Что такое прототипирование в разработке пользовательского интерфейса?
- A) Создание полной версии интерфейса до начала программирования
 - B) Создание базового макета интерфейса для оценки функциональности и дизайна
 - C) Удаление неиспользуемых элементов интерфейса
 - D) Изменение цветовой гаммы интерфейса в соответствии с требованиями клиента
14. Что такое макет пользовательского интерфейса?
- A) Описание дизайна и структуры интерфейса
 - B) Таблица, содержащая информацию обо всех элементах интерфейса
 - C) Список требований к функциональности интерфейса
 - D) Код, описывающий работу интерфейса
15. Какие основные элементы пользовательского интерфейса существуют?
- A) Кнопки, поля ввода, выпадающие списки
 - B) Базы данных, серверы, клиенты
 - C) Графики, звуки, анимация
 - D) Чаты, форумы, блоги
16. Что такое доступность интерфейса?
- A) Возможность интерфейса работать с разными операционными системами
 - B) Возможность использования интерфейса людьми с различными ограничениями
 - C) Возможность открытия и редактирования исходного кода интерфейса
 - D) Возможность удаленного доступа к интерфейсу
17. Какие технологии используются для создания графического пользовательского интерфейса?

- A) HTML и CSS
 - B) Python и Ruby
 - C) JavaScript и PHP
 - D) SQL и NoSQL
18. Что такое адаптивный дизайн интерфейса?
- A) Дизайн интерфейса, который изменяется в зависимости от времени суток
 - B) Дизайн интерфейса, который меняется в зависимости от местоположения пользователя
 - C) Дизайн интерфейса, который адаптируется к разным
19. Что такое UX-дизайн?
- A) Дизайн, ориентированный на создание приятного визуального образа
 - B) Дизайн, ориентированный на удобство использования и удовлетворение потребностей пользователя
 - C) Дизайн, ориентированный на создание максимально функциональных интерфейсов
 - D) Дизайн, ориентированный на создание эффективных и экономичных систем
20. Что такое контекстная помощь в пользовательском интерфейсе?
- A) Справочная система, предоставляющая информацию о том, как использовать элементы интерфейса
 - B) Кнопка, которая помогает пользователям поделиться приложением в социальных сетях
 - C) Элемент интерфейса, который позволяет пользователю сменить язык интерфейса
 - D) Всплывающее окно с предупреждением, что пользователь может потерять несохраненные данные при закрытии программы

Перечень заданий для оценки усвоенных умений

1. Разработайте прототип приложения, используя макеты интерфейса, цветовые схемы и шрифты, которые соответствуют заданным требованиям.
2. Напишите программу на языке программирования, которая отображает данные в таблице и позволяет пользователю взаимодействовать с ней, например, для редактирования или удаления строк.
3. Разработайте программу с использованием фреймворка для разработки мобильных приложений, которая имеет разные варианты макета интерфейса для разных устройств и разрешений экрана.
4. Напишите программу, используя библиотеку для создания графического интерфейса, которая позволяет пользователю рисовать на холсте с помощью мыши или сенсорного экрана.
5. Разработайте программу, которая отображает графики или диаграммы на основе данных, введенных пользователем или полученных из базы данных. Пользователь должен иметь возможность интерактивно управлять отображением данных, например, изменять масштаб или переключаться между различными представлениями.
6. Напишите программу на языке программирования, которая содержит узкие места в производительности. Проанализируйте и оптимизируйте код для ускорения работы программы.
7. Изучите код наследуемого проекта и выявите участки, которые можно улучшить и упростить, например, с помощью использования новых функций языка программирования или изменения алгоритма работы.
8. Разработайте скрипт или программу, которая автоматически выполняет рефакторинг кода, например, удаляет дублирующийся код, упрощает условные выражения или улучшает имена переменных.

9. Перепишите часть кода на другом языке программирования, который имеет лучшую производительность или поддержку для определенных функций, чтобы ускорить выполнение программы.
10. Проанализируйте производительность приложения и выявите проблемы, которые приводят к длительным задержкам или низкой отзывчивости. Разработайте план оптимизации, который позволит улучшить производительность приложения.

Экзамен (7 семестр)

Экзамен проводится по завершению изучения раздела МДК по окончании семестра в форме собеседования.

К сдаче экзамена допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации за семестр оценки не ниже «удовлетворительно».

Вопросы для подготовки к экзамену

Перечень вопросов для оценки усвоенных знаний

1. Что такое ADO.Net и для чего он используется?
2. Какие преимущества предоставляет ADO.Net в работе с базами данных?
3. Какие объекты ADO.Net используются для подключения к базе данных?
4. Какие объекты ADO.Net используются для выполнения запросов к базе данных?
5. Какие типы команд для работы с базой данных можно использовать в ADO.Net?
6. Что такое объектный тип доступа к данным в ADO.Net?
7. Какие технологии могут использоваться вместе с ADO.Net для работы с базой данных?
8. Какие роли и задачи могут быть назначены в команде, работающей с ADO.Net?
9. Какие методы тестирования ADO.Net могут использоваться?
10. Какие современные тренды в работе с базами данных с помощью ADO.Net существуют?
11. Что такое комплексная разработка программного обеспечения?
12. Какие этапы включает в себя комплексная разработка программного обеспечения?
13. Какие роли в команде разработки могут присутствовать при комплексной разработке ПО?
14. Какие основные методологии разработки программного обеспечения могут использоваться при комплексной разработке ПО?
15. Какие инструменты используются для управления процессом разработки ПО при комплексной разработке?
16. Какие методы тестирования используются при комплексной разработке ПО?
17. Каковы основные вызовы, связанные с комплексной разработкой ПО?
18. Какие технологии и языки программирования могут использоваться при комплексной разработке ПО?
19. Какие задачи могут решаться с помощью комплексной разработки ПО?
20. Каковы факторы, которые могут повлиять на успех комплексной разработки ПО?

Перечень заданий для оценки усвоенных умений

1. Разработайте общую архитектуру системы, определяющую ее структуру и основные компоненты.

2. Разработайте прототип системы, включающий в себя основные функции и интерфейс пользователя.
3. Напишите программу, которая обеспечивает взаимодействие различных модулей системы и обмен данными между ними.
4. Реализуйте функциональность системы, отвечающую за хранение и обработку данных, используя базу данных.
5. Напишите программу, которая обеспечивает безопасность системы, включая аутентификацию пользователей и контроль доступа к ресурсам.
6. Разработайте систему управления версиями кода, чтобы обеспечить эффективную разработку и сопровождение проекта.
7. Реализуйте автоматизированные тесты, которые позволяют проверить работоспособность и качество программного обеспечения.
8. Разработайте документацию системы, которая описывает ее структуру, функции, возможности и принципы работы.
9. Напишите программу, которая обеспечивает интеграцию системы с другими программными продуктами или сервисами.
10. Разработайте план сопровождения системы, который определяет процедуры обновления, исправления ошибок и поддержки пользователей.

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного учреждения
высшего образования
«Пермский национальный исследовательский политехнический университет»

ОЦЕНОЧНЫЕ СРЕДСТВА
МЕЖДИСЦИПЛИНАРНОГО КУРСА

МДК. 01.02 Поддержка и тестирование программных модулей

основной профессиональной образовательной программы
подготовки специалистов среднего звена
по специальности СПО 09.02.07 Информационные системы и программирование
(базовая подготовка)

Лысьва, 2023

ОБЩИЕ ПОЛОЖЕНИЯ

Контрольно – измерительные материалы по результатам изучения междисциплинарного курса **МДК. 01.02 Поддержка и тестирование программных модулей** ориентированы на проверку степени достижения требований к минимуму содержания и уровню подготовки обучающихся в соответствии с ФГОС и является основополагающим документом для организации контроля знаний, умений, практического опыта обучающихся в учебном процессе.

Результатом освоения междисциплинарного курса **МДК. 01.02 Поддержка и тестирование программных модулей** является:

Код и наименование профессиональных и общих компетенций, личностных результатов, формируемых в рамках ПМ	Основные показатели оценки результата
ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств	Уметь: — Выполнять отладку и тестирование программы на уровне модуля. — Оформлять документацию на программные средства. — Применять инструментальные средства отладки программного обеспечения. Знать: — Основные принципы отладки и тестирования программных продуктов — Инструментарий отладки программных продуктов
ПК 1.4 Выполнять тестирование программных модулей	Уметь: — Выполнять отладку и тестирование программы на уровне модуля. — Оформлять документацию на программные средства. Знать: — Основные виды и принципы тестирования программных продуктов.
ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода	Уметь: — Выполнять оптимизацию и рефакторинг программного кода. — Работать с системой контроля версий. Знать: — Способы оптимизации и приемы рефакторинга. — Инструментальные средства анализа алгоритма. — Методы организации рефакторинга и оптимизации кода. — Принципы работы с системой контроля версий.
ЛР 5	Демонстрирующий умение эффективно взаимодействовать в команде, вести диалог, в том числе с использованием средств коммуникации

<i>ЛР 6</i>	Демонстрирующий навыки анализа и интерпретации информации из различных источников с учетом нормативно-правовых норм
<i>ЛР 7</i>	Демонстрирующий готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности.
<i>ЛР 8</i>	Пользоваться профессиональной документацией на государственном и иностранном языках.
<i>ЛР 9</i>	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения
<i>ЛР 10</i>	Использовать знания по финансовой грамотности, планировать предпринимательскую деятельность в профессиональной сфере.
<i>ЛР 11</i>	Планировать и реализовывать собственное профессиональное и личностное развитие в условиях развития информационных технологий, применяемых в различных отраслях народного хозяйства.
<i>ЛР 12</i>	Активно применяющий полученные знания на практике
<i>ЛР 13</i>	Способный анализировать производственную ситуацию, быстро принимать решения
<i>ЛР 14</i>	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
<i>ЛР 17</i>	Проявлять доброжелательность к окружающим, деликатность, чувство такта и готовность оказать услугу каждому кто в ней нуждается.

1 МЕТОДЫ И ФОРМЫ КОНТРОЛЯ ОЦЕНИВАНИЯ ЭЛЕМЕНТОВ МЕЖДИСЦИПЛИНАРНОГО КУРСА

МДК. 01.02 Поддержка и тестирование программных модулей

1 Для текущего и рубежного контроля освоения дисциплинарных компетенций используются следующие методы:

- Устный опрос
- Наблюдение и оценка результатов практических занятий
- Наблюдение и оценка результатов лабораторных занятий
- Экспертная оценка результатов самостоятельной работы
- Экспертная оценка по результатам наблюдения за деятельностью обучающегося в

процессе освоения ПМ

2 Формой контроля промежуточной аттестации междисциплинарного курса являются: дифференцированный зачет (4семестр), экзамен (7 семестр), которые проводятся в сроки, установленные учебным планом и определяемые календарным учебным графиком образовательного процесса.

Таблица 1 – Методы и формы контроля и оценивания элементов междисциплинарного курса
МДК. 01.02 Поддержка и тестирование программных модулей

Элемент учебной дисциплины	Методы и формы контроля и оценивания		
	Текущий контроль	Рубежный контроль	Промежуточная аттестация
Раздел 1			
Тема 1.1 <i>Отладка и тестирование программного обеспечения</i>	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	<i>Тестирование</i>	<i>Дифференцированный зачет (6 семестр)</i>
Тема 1.2	Устный опрос	<i>Тестирование</i>	<i>Экзамен(7семестр)</i>

Документирование	Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ		
Форма контроля			Дифференцированный зачет Экзамен

Текущий контроль усвоения материала

Текущий контроль усвоения материала проводится в форме устного опроса студентов по темам дисциплины.

Наблюдение и оценка результатов практических занятий

Типовые темы практических занятий приведены в РПД. Комплект заданий на практические занятия приведены в МУ по ПЗ по учебной дисциплине.

Защита отчетов по практическим занятиям проводится индивидуально каждым обучающимся в форме собеседования.

Наблюдение и оценка результатов лабораторных занятий

Типовые темы лабораторных работ приведены в РПД. Комплект заданий на лабораторные занятия приведены в МУ по ЛР по учебной дисциплине.

Защита отчетов по лабораторным занятиям проводится индивидуально каждым обучающимся в форме собеседования.

Экспертная оценка результатов самостоятельной работы

Задания для самостоятельной работы приведены в МУ по СРС по учебной дисциплине.

Качественная оценка определения научного кругозора, степенью овладения методами теоретического исследования и развития самостоятельности мышления студента.

Способом проверки качества организации самостоятельной работы студентов является контроль:

- корректирующий (может осуществляться во время индивидуальных консультаций по поводу выполнения формы самостоятельной работы);
- констатирующий (по результатам выполнения специальных форм самостоятельной работы);
- самоконтроль (осуществляется самим студентом);
- текущий (в ходе выполнения различных форм самостоятельной работы, установленных рабочей программой);
- промежуточный (оценка результата обучения как итога выполнения студентом всех форм самостоятельной работы).

Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины

Осуществляется как наблюдение за процессом деятельности обучающегося в режиме реального времени. Является качественной оценкой освоения учебной дисциплины, учитываемой при промежуточной аттестации.

Рубежный контроль

Рубежный контроль для комплексного оценивания усвоенных знаний, освоенных умений проводится в форме, тестирования (после изучения разделов учебной дисциплины).

2 РЕЗУЛЬТАТЫ ОСВОЕНИЯ ЭЛЕМЕНТОВ МЕЖДИСЦИПЛИНАРНОГО КУРСА, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ ПРИ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

В результате промежуточной аттестации междисциплинарного курса осуществляется комплексная проверка следующих умений, знаний:

Результаты обучения (освоенные умения, усвоенные знания)	Показатели оценки результатов
Уметь:	
Выполнять отладку и тестирование программы на уровне модуля	Умеет выполнять отладку и тестирование программы на уровне модуля
Оформлять документацию на программные средства	Умеет оформлять документацию на программные средства
Применять инструментальные средства отладки программного обеспечения.	Умеет применять инструментальные средства отладки программного обеспечения
Выполнять отладку и тестирование программы на уровне модуля.	Умеет выполнять отладку и тестирование программы на уровне модуля
Оформлять документацию на программные средства	Умеет оформлять документацию на программные средства
Выполнять оптимизацию и рефакторинг программного кода	Умеет выполнять оптимизацию и рефакторинг программного кода
Работать с системой контроля версий	Умеет работать с системой контроля версий
Знать:	
Основные принципы отладки и тестирования программных продуктов	Знает основные принципы отладки и тестирования программных продуктов
Инструментарий отладки программных продуктов	Знает инструментарий отладки программных продуктов
Основные виды и принципы тестирования программных продуктов	Знает основные виды и принципы тестирования программных продуктов.
Способы оптимизации и приемы рефакторинга.	Знает способы оптимизации и приемы рефакторинга.
Инструментальные средства анализа алгоритма.	Знает инструментальные средства анализа алгоритма.
Методы организации рефакторинга и оптимизации кода.	Знает методы организации рефакторинга и оптимизации кода.
Принципы работы с системой контроля версий.	Знает принципы работы с системой контроля версий.

3 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ ЗНАНИЙ И УМЕНИЙ

МДК 01.02 Поддержка и тестирование программных модулей

Задания для оценки освоения Раздела 1 темы 1.1 Отладка и тестирование программного обеспечения

Обучающийся должен

знать:

- Основные принципы отладки и тестирования программных продуктов;
- Инструментарий отладки программных продуктов.
- Основные виды и принципы тестирования программных продуктов.
- Способы оптимизации и приемы рефакторинга
- Инструментальные средства анализа алгоритма
- Методы организации рефакторинга и оптимизации кода.
- Принципы работы с системой контроля версий.

уметь:

- Выполнять отладку и тестирование программы на уровне модуля;
- Оформлять документацию на программные средства;
- Применять инструментальные средства отладки программного обеспечения.
- Выполнять отладку и тестирование программы на уровне модуля;
- Оформлять документацию на программные средства;
- Выполнять оптимизацию и рефакторинг программного кода.
- Работать с системой контроля версий

Типовые вопросы для устного опроса

1. Унифицированный язык моделирования (UML). Определение прецедентов (вариантов использования). Построение концептуальной модели предметной области. Описание поведения системы с помощью диаграмм последовательности действий и диаграмм взаимодействий.

2. Проектирование программного обеспечения при структурном подходе. Структурная схема разрабатываемого программного обеспечения (логическая и физическая модели) Метод пошаговой детализации при составлении алгоритмов.

3. Основы файловой системы. Порядок работы с текстовыми и двоичными файлами. Функции для работы с файлами разного типа. Правила бесформатного и форматного ввода/вывода.

4. Проектирование программного обеспечения при объектном подходе. Структура проекта в объектно-ориентированной среде программирования. Событийно-управляемая модель программирования. Компоненты объектно-ориентированной среды. Принципы разработки графического интерфейса. Построение функциональной блок-схемы приложения. Особенности ввода/вывода в объектно-ориентированной среде программирования. Виды строк, функции преобразования типов.

5. Виды программных документов. Пояснительная записка. Руководство пользователя. Руководство системного программиста.

6. Тестирование методами «белого ящика» и «черного ящика». Порядок разработки тестов. Автоматизация тестирования. Модульное, интеграционное, системное тестирование.

7. Вставка, добавление и удаление записей таблиц базы данных. Поиск, фильтрация и сортировка данных. Возможности среды для создания новых полей.

Типовой тест

Тема 1.1 Основные методы внедрения и анализа функционирования программного обеспечения

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;

- тест выполняется *в электронном виде с помощью Onlinetestpad.com*

Инструкция: на выполнение теста отводится 30 минут, внимательно прочитайте вопрос, выберите один вариант ответа и заносится в бланк ответов

Контрольные вопросы для проведения тестирования

Тест 1: Модульное и интеграционное тестирование Список вопросов теста

1. Какие существуют особенности управляющего графа программы в случае объектно-ориентированного программирования (ООП)?

- a. она становится неприменимой
- b. она требует адаптации по обработке сообщений
- c. она требует описания поведения программы
- d. она требует описания не только структуры, но и поведения программы

2. . Какие существуют особенности интеграционного тестирования для ООП?

- a. тестирование дерева классов
- b. тестирование последовательностей прямых вызовов методов с помощью Rпутей
- c. тестирование последовательностей обработки сообщений с помощью MMпутей
- d. тестирование исключительно цепочек вызовов процедур программ

3. Какие этапы включает методика ООП при тестировании программного комплекса?

- a. тестирование методов каждого класса программного комплекса
- b. тестирование отношений между классами с помощью тестов на основе Rпутей или MM-путей
- c. тестирование взаимодействия модулей по всей иерархии комплекса

4. Какие этапы методики тестирования используются в ООП?

- a. тестирование методов каждого класса программы
- b. тестирование методов класса, входящих в его контекст
- c. тестирование дерева классов программного проекта, включающего оттестированный класс

5. . Какие этапы методов тестирования класса используются в ООП?

- a. тестирование класса как модуля по выбранному критерию
- b. тестирование класса как иерархической структуры
- c. тестирование классов, входящих в модель проекта

6. Что такое управляющий граф программы (УГП)?

- a. множество операторов программы
 - b. граф, вершины которого кодируют операторы программы, а дуги - управления (порядок исполнения) операторов
 - c. множество операторов управления
- 7. Что такое путь в УГП?**
- a. последовательность вершин и дуг управляющего графа программы с фиксированными начальной и конечной вершиной
 - b. последовательность ветвей управляющего графа программы с фиксированными начальной вершиной первой ветви и конечной вершиной последней ветви пути
 - c. множество связанных дуг управляющего графа программы
- 8. Что такое ветвь управляющего графа программы?**
- a. последовательность вершин и дуг управляющего графа программы с фиксированными начальной и конечной вершиной, которые кодируют либо условные операторы, либо первый и последний операторы управляющего графа программы соответственно
 - b. часть пути, в котором все внутренние вершины кодируют линейные операторы
 - c. начальная и конечная вершина пути
- 9. Можно ли гарантировать остановку программы на любом тесте?**
- a. В общем случае нет
 - b. возможно в частных случаях
 - c. задача в общей постановке алгоритмически неразрешима
- 10. Какие задачи у модульного тестирования?**
- a. выявление локальных ошибок реализации алгоритмов модулей
 - b. выявление ошибок при вызове модулей
 - c. выявление ошибок взаимодействия модуля с окружением
- 11. На основе каких принципов строятся тесты для модульного тестирования?**
- a. анализ потоков управления модуля
 - b. анализ потоков данных модуля
 - c. анализ покрытия в соответствии с заданными критериями C0, C1, C2
- 12. Каковы фазы процесса построения тестовых путей?**
- a. построение управляющего графа программы
 - b. выбор тестовых путей
 - c. генерация тестов, соответствующих выбранным тестовым путям
- 13. Какие существуют методы построения тестовых путей?**
- a. статические
 - b. динамические
 - c. методы реализуемых путей
- 14. Как реализуются динамические методы построения тестовых путей?**
- a. наращивание начальных отрезков реализованных путей продолжающими их фрагментами, чтобы увеличить покрытие
 - b. построение пути методом удлинения за счет добавления дуг
 - c. поиск всех реализуемых путей
- 15. Какие существуют разновидности интеграционного тестирования?**
- a. монолитное тестирование
 - b. нисходящее тестирование
 - c. восходящее тестирование

- d. Регрессионное тестирование
- 16. **Каковы особенности нисходящего тестирования?**
 - a. необходимость разработки заглушек
 - b. необходимость разработки среды управления очередностью вызовов модулей
 - c. параллельная разработка эффективных модулей
- 17. **Каковы особенности восходящего тестирования?**
 - a. минимизация разработки заглушек
 - b. запаздывание в проверке функциональности реализуемого приложения
 - c. необходимость разработки среды управления очередностью вызовов модулей
- 18. **В чем заключаются особенности интеграционного тестирования для процедурного программирования?**
 - a. тестирование программных комплексов, заданных в виде иерархических структур модулей
 - b. использование диаграмм потока управления в качестве модели тестируемого комплекса
 - c. контроль соответствия спецификациям параметров модулей и межмодульных связей
 - d. контроль наследования
- 19. **При создании очередной версии программы была добавлена функция А, функция D была удалена, функция С – изменена. К какой группе относится тест, покрывающий только функцию А?**
 - a. тесты, пригодные для повторного использования
 - b. тесты, требующие повторного запуска
 - c. устаревшие тесты
 - d. новые тесты
- 20. **Возможно ли тестирование программы на всех допустимых значениях параметров?**
 - a. никогда
 - b. да, всегда
 - c. возможно в отдельных случаях

Тест 2: Системное и регрессионное тестирование Список вопросов теста

- 1. **Каковы особенности системного тестирования?**
 - a. тестированию подлежит система в целом
 - b. тесты оперируют пользовательским или другими внешними интерфейсами
 - c. структура проекта тестируется на уровне подсистем
 - d. тестирование осуществляется по методу «черного ящика»
- 2. **Какие задачи решаются на уровне системного тестирования?**
 - a. выявление дефектов в функционировании приложения или в работе с ним
 - b. выявление дефектов использования ресурсов
 - c. выявление несовместимости с окружением
 - d. выявление непредусмотренных сценариев применения
 - e. использования непредусмотренных комбинаций данных
- 3. **Какие категории тестов разрабатываются для системного тестирования?**
 - a. тесты для проверки полноты функциональности

- b. тесты для проверки корректности использования ресурсов
- c. тесты для проверки стрессовых режимов и оценки производительности
- d. тесты для проверки защиты от искаженных данных и некорректных действий
- e. тесты для проверки инсталляции и конфигурации для платформ, предусмотренных спецификацией
- f. тесты для проверки корректности пользовательской документации

4. Каковы особенности регрессионного тестирования?

- a. регрессионное тестирование является подмножеством системного тестирования
- b. выбор между полным и частичным перетестированием и пополнением тестовых наборов
- c. перетестирование предусматривает только контроль частей приложения, связанных с изменениями

5. Какие задачи решает тестировщик, проверяя изменения, внесенные разработчиком в код?

- a. проверка и подтверждение исправления дефекта
- b. обеспечение гарантий, что качество исправленного кода не ухудшилось
- c. перетестирование последствий исправлений на предмет обнаружения индуцированных ошибок
- d. проверка возможности воспроизведения ошибки каким-либо другим способом

6. Какие типы дефектов выявляются при системном или при регрессионном тестировании?

- a. отсутствующая или некорректная функциональность
- b. некорректность проектной документации
- c. непредусмотренные данные или неподдерживаемые сценарии использования
- d. ошибки переносимости на другие платформы
- e. ошибки пользовательской документации
- f. ошибки инсталляции и конфигурирования

7. Можно ли гарантировать безопасность метода регрессионного тестирования при отсутствии информации об изменениях в программе

- a. нет
- b. да

8. Какие методы регрессионного тестирования применяются в условиях высоких требований к качеству программного продукта?

- a. метод повторного прогона всех тестов
- b. случайные методы
- c. безопасные методы
- d. методы минимизации
- e. методы, основанные на покрытии кода

9. Какими свойствами обладает метод повторного прогона всех тестов?

- a. полнота
- b. точность
- c. Эффективность
- d. универсальность

10. Можно ли гарантировать безопасность метода регрессионного тестирования в условиях отсутствия тестов, использовавшихся при тестировании предыдущих версий?

- a. тесты, пригодные для повторного использования

- b. тесты, требующие повторного запуска
 - c. устаревшие тесты
 - d. новые тесты
11. **Какие из перечисленных методов тестирования наиболее затратные**
- a. системное тестирование с моделируемым окружением
 - b. статические методы тестирования
 - c. интеграционное тестирование
 - d. системное тестирование в реальном окружении и реальном времени
 - e. модульное тестирование
12. **Какие из перечисленных методов тестирования дают наиболее надежные результаты?**
- a. статические методы
 - b. модульное тестирование
 - c. интеграционное тестирование
 - d. системное тестирование с моделируемым окружением
 - e. системное тестирование в реальном окружении и реальном времени
13. **Что влияет на обеспечение качества программного продукта?**
- a. тестирование
 - b. анализ дизайна
 - c. обзоры кода
 - d. аудиты процесса разработки ревьюирование проектной документации
 - e. ревьюирование проектной документации
14. **Какие основные уровни выделены в процессе тестирования?**
- a. модульное тестирование
 - b. Регрессионное тестирование
 - c. интеграционное тестирование
 - d. системное тестирование
15. **Какие методы регрессионного тестирования применяются в условиях высоких требований к программному продукту?**
- a. метод повторного прогона всех тестов
 - b. случайные методы
 - c. безопасные методы
 - d. методы, основанные на покрытии кода
 - e. методы минимизации
16. **В среднем, метод выборочного регрессионного тестирования...**
- a. обнаруживает столько же ошибок, сколько и метод повторного прогона всех тестов
 - b. обнаруживает больше ошибок, чем метод повторного прогона всех тестов
 - c. обнаруживает меньше ошибок, чем метод повторного прогона всех тестов
 - d. может обнаруживать больше или меньше ошибок, чем метод повторного прогона всех тестов
17. **На каком этапе регрессионного тестирования удаляются устаревшие тесты?**
- a. предсказание целесообразности
 - b. идентификация изменений
 - c. отбор тестов
 - d. выполнение тестов

- e. создание дополнительных тестов
- f. обновление базы данных

18. На каком этапе регрессионного тестирования проводится упорядочение тестов?

- a. предсказание целесообразности
- b. идентификация изменений
- c. отбор тестов
- d. выполнение тестов
- e. создание дополнительных тестов
- f. обновление базы данных

Тема 1.2 Документирование

Типовые вопросы для устного опроса

1. Какую роль играет документирование кода в разработке программного обеспечения?
2. Какие виды комментариев в коде вы знаете и как их правильно использовать?
3. Каковы преимущества использования специальных инструментов для документирования кода?
4. Какие основные элементы должны присутствовать в документации функции?
5. Какие принципы должны быть соблюдены при написании документации для кода?
6. Какие средства документирования используются в языке программирования Python?
7. Что такое документирование API и как оно связано с документированием кода?
8. Как можно улучшить качество документации кода?
9. Каковы риски отсутствия документации в программном коде?
10. Какие инструменты и методы документирования кода вы бы порекомендовали новичкам в программировании?
11. Какую роль играет документация в жизненном цикле разработки программного продукта?
12. Какие виды документов используются в процессе разработки программного продукта?
13. Какова структура технического задания на разработку программного продукта?
14. Какие требования к документации устанавливаются в стандартах качества программного обеспечения?
15. Какие основные элементы должны быть задокументированы в спецификации требований на разработку программного продукта?
16. Какие этапы включает процесс документирования программного продукта?
17. Какие принципы должны быть соблюдены при написании технической документации на программный продукт?
18. Каковы преимущества использования специальных инструментов для документирования программного продукта?
19. Каковы риски отсутствия документации на программный продукт?
20. Какие инструменты и методы документирования программного продукта вы бы порекомендовали новичкам в разработке программного обеспечения?

Типовой тест

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью [Onlinetestpad.com](https://onlinetestpad.com)

Инструкция: на выполнение теста отводится 30 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

Вариант 1

1. Какой тип документации содержит описание требований к программному продукту?
 - a) Технический план
 - b) Техническое задание
 - c) Руководство пользователя
 - d) Тест-кейсы
2. Какая документация содержит информацию о процессах, методах и инструментах, используемых при разработке программного продукта?
 - a) Технический план
 - b) Техническое задание
 - c) Руководство разработчика
 - d) Руководство пользователя
3. Какая документация содержит информацию о тестировании программного продукта?
 - a) Технический план
 - b) Техническое задание
 - c) Руководство тестировщика
 - d) Руководство пользователя
4. Какая документация описывает структуру и функциональность программного продукта?
 - a) Технический план
 - b) Техническое задание
 - c) Руководство разработчика
 - d) Руководство пользователя
5. Какая документация содержит информацию о принципах работы программного продукта?
 - a) Технический план
 - b) Техническое задание
 - c) Руководство разработчика
 - d) Руководство пользователя
6. Какая документация описывает процесс установки программного продукта?
 - a) Технический план
 - b) Техническое задание
 - c) Руководство по установке
 - d) Руководство пользователя
7. Какая документация содержит информацию о конфигурации и настройках программного продукта?
 - a) Технический план
 - b) Техническое задание
 - c) Руководство по установке
 - d) Руководство пользователя
8. Какая документация содержит информацию о совместимости программного продукта с другими программными продуктами?
 - a) Технический план
 - b) Техническое задание

- c) Руководство по установке
 - d) Руководство пользователя
9. Какая документация содержит информацию о том, как использовать функционал программного продукта?
- a) Технический план
 - b) Техническое задание
 - c) Руководство разработчика
 - d) Руководство по эксплуатации
10. Какая документация описывает порядок взаимодействия пользователя с программным продуктом?
- a) Технический план
 - b) Техническое задание
 - c) Руководство пользователя
 - d) Руководство разработчика

4 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

МДК. 01.02 Поддержка и тестирование программных модулей

Изучение *МДК. 01.02 Поддержка и тестирование программных модулей* реализуется в течение 2 семестров.

Формами контроля промежуточной аттестации *МДК. 01.02 Поддержка и тестирование программных модулей* является дифференцированный зачет в 6 семестре, экзамен в 7 семестре.

Основой для определения оценки при проведении промежуточных аттестаций служит объем и уровень усвоения обучающимися материала, предусмотренного рабочей программой профессионального модуля *ПМ.01 Разработка модулей программного обеспечения для компьютерных систем* в части *МДК. 01.02 Поддержка и тестирование программных модулей*.

Дифференцированный зачет (6 семестр)

Дифференцированный зачет проводится по завершению изучения раздела МДК по окончании семестра в форме теста с учетом результатов текущего контроля.

К сдаче дифференцированного зачета допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации за семестр оценки не ниже «удовлетворительно».

Вопросы для подготовки к дифференцированному зачету

1. Что такое отладка программного обеспечения, и почему она важна?
2. Какие инструменты отладки вы знаете, и для чего они используются?
3. Какие типы тестирования программного обеспечения существуют?
4. Что такое unit -тестирование, и как оно связано с отладкой?
5. Какой процесс следует за тестированием программного обеспечения?
6. Какие методы тестирования безопасности программного обеспечения вы знаете?
7. Что такое регрессионное тестирование, и как оно помогает в отладке?
8. Как вы можете определить, что программа не работает должным образом?
9. Какие преимущества обеспечивает автоматизированное тестирование?
10. Какие типы ошибок могут возникнуть в программном обеспечении, и как они могут быть обнаружены?
11. Как оценить качество тестирования ПО?
12. Какие проблемы могут возникнуть при отладке и тестировании ПО и как их решать?
13. Каким образом можно проводить тестирование пользовательского интерфейса ПО?
14. Расскажите про методы тестирования совместимости ПО с разными операционными системами и браузерами?
15. Какие существуют подходы к созданию тестовых сценариев и как их выбирать?
16. Какие критерии выбора тестовых данных следует учитывать при тестировании ПО?

Перечень заданий для оценки усвоенных умений

1. Написать unit-тесты для функции, которая возвращает сумму двух чисел.
2. Написать функциональные тесты для веб-приложения, проверяющие корректность регистрации нового пользователя.
3. Создать набор тестов для проверки производительности приложения в условиях высокой нагрузки.
4. Отладить код функции, которая не корректно считает факториал числа, и исправить ее.
5. Проанализировать логи ошибок приложения и найти причину возникновения конкретной ошибки.
6. Провести тестирование совместимости приложения с разными версиями операционных систем и браузеров.
7. Написать тесты для проверки безопасности приложения, в том числе на уязвимости типа SQL-инъекций или XSS-атак.
8. Создать набор тестов для проверки корректности работы приложения на разных языках и в разных локализациях.
9. Написать тесты для проверки работы алгоритма сортировки на больших объемах данных.
10. Исправить баг в приложении, который приводит к неправильному отображению данных на графике, и проверить его исправление при помощи регрессионного тестирования.

Экзамен (7 семестр)

Экзамен проводится по завершению курса изучения МДК в форме теста.

К сдаче экзамена допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации за семестр оценки не ниже «удовлетворительно».

Вопросы для подготовки к экзамену

Перечень вопросов для оценки усвоенных знаний

1. Введение: тестирование - способ обеспечения качества программного продукта
2. Основные понятия тестирования
3. Подходы к обоснованию истинности формул и программ и их связь с тестированием.

Вопросы организации тестирования. Фазы тестирования, основные проблемы тестирования и поставлена задача выбора конечного набора тестов.

4. Требования к идеальному критерию тестирования и классы частных критериев. Особенности применения структурных и функциональных критериев на базе конкретных примеров. Особенности применения методов стохастического тестирования и метод оценки скорости выявления ошибок. Мутационный критерий и на примере иллюстрируется техника работы с ним.

5. Оценка оттестированности проекта: метрики и методика интегральной оценки

6. Графовые модели проекта, метрики оценки оттестированности проекта, приводятся примеры плоской и иерархической моделей проекта.

7. Особенности модульного тестирования, подходы к тестированию на основе потока управления, потока данных, динамические и статические методы при структурном подходе. Взаимосвязь сборки модулей и методов интеграционного тестирования. Подходы монолитного, инкрементального, нисходящего и восходящего тестирования. Рассматриваются особенности интеграционного тестирования в процедурном программировании.

8. Интеграционное тестирование и его особенности для объектноориентированного программирования

9. Модель объектно-ориентированной программы. Оценки сложности тестирования и методика тестирования объектно-ориентированной программы. Рассматривается пример интеграционного тестирования.

10. Разновидности тестирования: системное и регрессионное тестирование

11. Автоматизация тестирования

12. Особенности индустриального тестирования

13. Документирование и оценка индустриального тестирования

14. Описываются особенности документирования тестовых процедур для ручных и автоматизированных тестов, описаний тестовых наборов и тестовых отчетов. Рассматривается жизненный цикл дефекта. Обсуждаются метрики, используемые при тестировании.

15. Регрессионное тестирование: цели и задачи, условия применения, классификация тестов и методов отбора

16. Регрессионное тестирование: разновидности метода отбора тестов

17. Регрессионное тестирование: методики, не связанные с отбором тестов и методики порождения тестов

18. Регрессионное тестирование: алгоритм и программная система поддержки

19. Описание тестируемой системы и ее окружения. Планирование тестирования

20. Модульное тестирование на примере классов

21. Интеграционное тестирование

22. Системное тестирование

23. Ручное тестирование

24. Автоматизация тестирования с помощью скриптов

25. Автоматическая генерация тестов на основе формального описания

26. Описание ручного тестирования

27. Автоматизация тестирования с помощью скриптов

28. Описание автоматической генерации MSC тестов

Перечень заданий для оценки усвоенных умений

1. Определить функции менеджера сопровождения и менеджера развертывания в рамках работы администратора базы данных (БД). Указать регламенты и процедуры установки и настройки программного обеспечения в рамках работы администратора БД, обеспечивающие функционирование БД.
2. На основе ГОСТ 19.505-79 разработать сборник рекомендаций по обучению персонала правилам эксплуатации отраслевого программного обеспечения
3. Определить основные задачи, решаемые программным обеспечением (ПО) для поддержания работы пользователей с базой данных (БД), Задать методику и параметры, требуемые для настройки ПО с целью обеспечения работы пользователей с базой данных
4. Определить основные процессы и документы, стандартизованные по ГОСТ Р ИСО/МЭК 12207 и ГОСТ Р 57193-2016 для информационной системы, дать их основные характеристики и взаимосвязь в рамках информационной системы

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного учреждения
высшего образования
«Пермский национальный исследовательский политехнический университет»

ОЦЕНОЧНЫЕ СРЕДСТВА

МЕЖДИСЦИПЛИНАРНОГО КУРСА

МДК 01.03 Разработка мобильных приложений

основной профессиональной образовательной программы
подготовки специалистов среднего звена
по специальности СПО 09.02.07 Информационные системы и программирование
(базовая подготовка)

Лысьва, 2023

ОБЩИЕ ПОЛОЖЕНИЯ

Контрольно – измерительные материалы по результатам изучения междисциплинарного курса **МДК 01.03 Разработка мобильных приложений** ориентированы на проверку степени достижения требований к минимуму содержания и уровню подготовки обучающихся в соответствии с ФГОС и является основополагающим документом для организации контроля знаний, умений, практического опыта обучающихся в учебном процессе.

Результатом освоения междисциплинарного курса **МДК 01.03 Разработка мобильных приложений** является:

Код и наименование профессиональных и общих компетенций, личностных результатов, формируемых в рамках ПМ	Основные показатели оценки результата
ПК 1.2 <i>Разрабатывать программные модули в соответствии с техническим заданием</i>	Уметь: – Создавать программу по разработанному алгоритму как отдельный модуль.; – Оформлять документацию на программные средства; —Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней, в том числе для мобильных платформ Знать: – Основные этапы разработки программного обеспечения.; —Основные принципы технологии структурного и объектно-ориентированного программирования —Знание API современных мобильных операционных систем
ПК 1.6 <i>Разрабатывать модули программного обеспечения для мобильных платформ</i>	Уметь: – Осуществлять разработку кода программного модуля на современных языках программирования – Оформлять документацию на программные средства Знать: – Основные этапы разработки программного обеспечения. – Основные принципы технологии структурного и объектно-ориентированного программирования.
ЛР 5	Демонстрирующий умение эффективно взаимодействовать в команде, вести диалог, в том числе с использованием средств коммуникации
ЛР 6	Демонстрирующий навыки анализа и интерпретации информации из различных источников с учетом нормативно-правовых норм
ЛР 7	Демонстрирующий готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности.

<i>ЛР 8</i>	Пользоваться профессиональной документацией на государственном и иностранном языках.
<i>ЛР 9</i>	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения
<i>ЛР 10</i>	Использовать знания по финансовой грамотности, планировать предпринимательскую деятельность в профессиональной сфере.
<i>ЛР 11</i>	Планировать и реализовывать собственное профессиональное и личностное развитие в условиях развития информационных технологий, применяемых в различных отраслях народного хозяйства.
<i>ЛР 12</i>	Активно применяющий полученные знания на практике
<i>ЛР 13</i>	Способный анализировать производственную ситуацию, быстро принимать решения
<i>ЛР 14</i>	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
<i>ЛР 17</i>	Проявлять доброжелательность к окружающим, деликатность, чувство такта и готовность оказать услугу каждому кто в ней нуждается.

1 МЕТОДЫ И ФОРМЫ КОНТРОЛЯ ОЦЕНИВАНИЯ ЭЛЕМЕНТОВ МЕЖДИСЦИПЛИНАРНОГО КУРСА

МДК 01.03 Разработка мобильных приложений

1 Для текущего и рубежного контроля освоения дисциплинарных компетенций используются следующие методы:

- Устный опрос
- Наблюдение и оценка результатов практических занятий
- Наблюдение и оценка результатов лабораторных занятий
- Наблюдение и оценка результатов курсового проектирования
- Экспертная оценка результатов самостоятельной работы
- Экспертная оценка по результатам наблюдения за деятельностью обучающегося в

процессе освоения ПМ

2 Формой контроля промежуточной аттестации междисциплинарного курса являются: экзамен (5 семестр), который проводится в сроки, установленные учебным планом и определяемые календарным учебным графиком образовательного процесса.

Таблица 1 – Методы и формы контроля и оценивания элементов междисциплинарного курса
МДК 01.03 Разработка мобильных приложений

Элемент учебной дисциплины	Методы и формы контроля и оценивания		
	Текущий контроль	Рубежный контроль	Промежуточная аттестация
Раздел 1 Разработка мобильных приложений			
Тема 1.1 Основные платформы и языки разработки мобильных приложений	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	<i>Тестирование</i>	Экзамен (5 семестр)
Тема 1.2 Создание и	Устный опрос Наблюдение и	<i>Тестирование</i>	

тестирование модулей для мобильных приложений	оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ		
Курсовой проект (работа)	Наблюдение и оценка результатов курсового проекта (работы)	<i>Защита курсового проекта (работы)</i>	
Форма контроля			Экзамен

Текущий контроль усвоения материала

Текущий контроль усвоения материала проводится в форме устного опроса студентов по темам дисциплины.

Наблюдение и оценка результатов практических занятий

Типовые темы практических занятий приведены в РПД. Комплект заданий на практические занятия приведены в МУ по ПЗ по учебной дисциплине.

Защита отчетов по практическим занятиям проводится индивидуально каждым обучающимся в форме собеседования.

Наблюдение и оценка результатов лабораторных занятий

Типовые темы лабораторных работ приведены в РПД. Комплект заданий на лабораторные занятия приведены в МУ по ЛР по учебной дисциплине.

Защита отчетов по лабораторным занятиям проводится индивидуально каждым обучающимся в форме собеседования.

Наблюдение и оценка результатов курсового проекта (работы)

Типовые темы для разработки курсового проекта (работы) приведены в РП ПМ. Комплект заданий и указания к выполнению курсового проекта (работы) приведены в МУ по КП.

Защита курсового проекта (работы) проводится индивидуально каждым обучающимся в форме собеседования.

Экспертная оценка результатов самостоятельной работы

Задания для самостоятельной работы приведены в МУ по СРС по учебной дисциплине.

Качественная оценка определения научного кругозора, степенью овладения методами теоретического исследования и развития самостоятельности мышления студента.

Способом проверки качества организации самостоятельной работы студентов является контроль:

- корректирующий (может осуществляться во время индивидуальных консультаций по поводу выполнения формы самостоятельной работы);
- констатирующий (по результатам выполнения специальных форм самостоятельной работы);
- самоконтроль (осуществляется самим студентом);
- текущий (в ходе выполнения различных форм самостоятельной работы, установленных рабочей программой);
- промежуточный (оценка результата обучения как итога выполнения студентом всех форм самостоятельной работы).

Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины

Осуществляется как наблюдение за процессом деятельности обучающегося в режиме реального времени. Является качественной оценкой освоения учебной дисциплины, учитываемой при промежуточной аттестации.

Рубежный контроль

Рубежный контроль для комплексного оценивания усвоенных знаний, освоенных умений проводится в форме, тестирования (после изучения разделов учебной дисциплины).

2 РЕЗУЛЬТАТЫ ОСВОЕНИЯ ЭЛЕМЕНТОВ МЕЖДИСЦИПЛИНАРНОГО КУРСА, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ ПРИ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

В результате промежуточной аттестации междисциплинарного курса осуществляется комплексная проверка следующих умений, знаний:

Результаты обучения (освоенные умения, усвоенные знания)	Показатели оценки результатов
Уметь:	
Создавать программу по разработанному алгоритму как отдельный модуль.;	Умеет создавать программу по разработанному алгоритму как отдельный модуль
Оформлять документацию на программные средства;	Умеет оформлять документацию на программные средства
Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней, в том числе для мобильных платформ	Умеет осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней, в том числе для мобильных платформ
Осуществлять разработку кода программного модуля на современных языках программирования	Умеет осуществлять разработку кода программного модуля на современных языках программирования
Знать:	
Основные этапы разработки программного обеспечения	Знает основные этапы разработки программного обеспечения
Основные принципы технологии структурного и объектно-ориентированного программирования	Знает основные принципы технологии структурного и объектно-ориентированного программирования
Знание API современных мобильных операционных систем	Знает знание API современных мобильных операционных систем

3 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ ЗНАНИЙ И УМЕНИЙ

МДК 01.03 Разработка мобильных приложений

Задания для оценки освоения Раздела 1

темы 1.1 Основные платформы и языки разработки мобильных приложений

Обучающийся должен

знать:

- Основные этапы разработки программного обеспечения
- Основные принципы технологии структурного и объектно-ориентированного программирования
- Знание API современных мобильных операционных систем

уметь:

- Создавать программу по разработанному алгоритму как отдельный модуль.;
- Оформлять документацию на программные средства;
- Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней, в том числе для мобильных платформ
- Осуществлять разработку кода программного модуля на современных языках программирования

Типовые вопросы для устного опроса

1. Архитектура платформы Android. Уровень ядра. Уровень библиотек.
2. Архитектура платформы Android. Dalvik Virtual Machine.
3. Архитектура платформы Android. Уровень каркаса приложений. Уровень приложений.
4. Среда разработки для Android. Eclipse IDE. Плагин ADT. Android Virtual Device.
5. Android SDK. Версии SDK и Android API Level.
6. Структура проекта Android-приложения в Eclipse. Каталоги ресурсов. Файл R.java.
7. Графический интерфейс пользователя в Android-приложениях. XMLpa3MeiKa интерфейса.
8. Архитектура платформы Android.
9. XML .-разметка интерфейса пользователя.
10. XAML .-разметка интерфейса пользователя.
11. Базовые элементы управления.
12. Ресурсы в Android-приложениях.
13. Ресурсы в Windows Phone-приложениях.
14. Активности и интенты.
15. Обработка пользовательского ввода. Касания, ввод текста.
16. Типы компоновок графического интерфейса. FraneLayout, LinearLayout, TableLayout, RelativeLayout.
17. Базовые элементы управления. TextView. EditText. Тип ввода текста. Параметры отображения клавиатуры. ImageView.
18. Диалоговые окна. AlertDialog. ProgressDialog. DatePickerDialog. TimePickerDialog. Создание пользовательских диалоговых окон.
19. Многопоточные приложения в Android и Windows Phone. Использование системных таймеров и системного времени.

20. Процессы в Android. Объекты Activity. Состояния Activity.
21. Использование объектов Intent. Intent-фильтры.
22. 22. Использование ресурсов. Ссылки на ресурсы. Загрузка простых типов из ресурсов. Загрузка файлов произвольного типа.
23. Файловая система Android. Чтение и запись файлов.
24. Адаптеры данных. Отображение данных в компонентах ListView, GridView, AutoCompleteTextView, MultiAutoCompleteTextView.
25. Пользовательские настройки. Использование SharedPreferences. Виды настроек.
26. Работа с графикой. Drawable и Canvas.
27. Работа с анимацией. Tween Animation и Frame Animation. Описание анимации в XML и в коде программы.
28. Службы в Android. Компонент Service.
29. Датчики мобильных устройств. Управление датчиками в приложении. Виды датчиков и особенности их использования.
30. Программный доступ к дисплею устройства. Менеджер окон. Параметры дисплея.

Контрольные вопросы для проведения тестирования

Тема 1.1 Основные платформы и языки разработки мобильных приложений

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется **в электронном виде с помощью Onlinetestpad.com**

Инструкция: на выполнение теста отводится 30 минут, внимательно прочитайте вопрос, выберите один вариант ответа и заносится в бланк ответов

Контрольные вопросы для проведения тестирования

1. Какие методы отражают жизненный цикл Activity:

- a. onCreate() (+)
- b. onRestart()(+)
- c. onResume()(+)
- d. onRunning()
- e. onClose()

2. Подробная информация о приложении содержится в файле:

- a. AndroidManifest.xml (+)
- b. main.xml
- c. R.java
- d. default.properties

3. Для создания всплывающего уведомления необходимо инициализировать объект:

- a. Toast (+)
- b. Message
- c. TextView
- d. MessageBox

4. Единица измерения dp или dip - это 1/72 дюйма,

- a. определяется по физическому размеру экрана дюйм,

- b. определяется по физическому размеру экрана абстрактная ЕИ,
- c. позволяющая приложениям выглядеть одинаково на различных экранах и разрешениях (+)
- d. физический элемент матрицы дисплея

5. Тип верстки при котором позиционирование элементов происходит относительно друг друга и относительно главного контейнера

- a. AbsoluteLayout
- b. FrameLayout
- c. LinearLayout
- d. RelativeLayout (+)

6. Какой класс можно использовать для перехода между Activity?

- a. ActivityChanger
- b. Activity Intent (+)
- c. Switcher

7. От какого класса наследуются все элементы управления?

- a. Control
- b. Controls
- c. Element
- d. View (+)

8. Для чего используется класс R?

- a. Класс ресурсов (+)
- b. Класс для управления элементами управления
- c. Класс для работы с потоками
- d. Класс, предназначенный только для доступа к идентификаторам элементов

9. Как программно получить доступ к элементам управления через идентификатор?

- a. getViewById(id)
- b. findViewById(id) (+)
- c. findView(id)
- d. getView(id)

10. В какой библиотеке находится класс Activity?

- a. android
- b. android.Activities
- c. android.app (+)
- d. java.app

11. На основе какой операционной системы была разработана система Android:

- a. Linux (+)
- b. MiniX
- c. Windows
- d. iOS

12. Как называется режим использования Android-системы с максимальными правами (аналог администратора Windows):

- a. Суперпользователь (+)
- b. Root-Home пользователь
- c. Android user
- d. Up-User

13. Как называется событие-клик объекта-кнопки:

- a. OnClick (+)
- b. OnClicked
- c. Click
- d. ClickEvent

14. Для добавления в приложение более одного экрана взаимодействия необходимо создание:

- a. Activity (+)
- b. View
- c. Layout
- d. XML-файла

15. Как называется класс для работы с SQLiteбдами в Android:

- a. SQLiteOpenHelper (+)
- b. SQLProvider
- c. SQLiteProvider
- d. SQLiteContext

16. Как называется язык разметки Window Phone приложений:

- a. XAML (+)
- b. XML
- c. Java
- d. C#

17. С помощью какого класса можно вывести сообщение в Window Phone приложении:

- a. Toast
- b. MessageBox (+)
- c. Message
- d. ShowClass

18. Какой элемент предоставляет вариант разметки, элементы, помещенные в который, располагаются в стек горизонтально или вертикально:

- a. StackPanel (+)
- b. Pivot
- c. Panorama
- d. Canvas

19. Какой элемент позволяют позиционировать элементы внутри себя максимально гибко предоставляя возможность размещать элементы, используя строки и столбцы.

- a. Grid (+)
- b. Pivot
- c. Panorama
- d. Canvas

20. Какой элемент предоставляет наиболее простой вариант разметки. Он может быть использован для абсолютного позиционирования элементов с использованием координат.

- a. Grid
- b. Pivot
- c. Panorama
- d. Canvas (+)

21. На базе какого ядра сделано ядро Android?

- a. Windows NT
 - b. Linux
 - c. FreeBSD
 - d. VxWorks
- 22. Какое название носит виртуальная Java-машина Android?**
- a. Dalvik
 - b. B ionic
 - c. HotSpot
- 23. Как изменился набор консольных утилит Linux в Android?**
- a. Существенно уменьшился
 - b. Остался без изменений
 - c. Существенно увеличился
 - d. В Android нет консольных утилит
- 24. Для чего предназначен Binder?**
- a. Связывает файлы ресурсов с приложением
 - b. Обеспечивает межпроцессное взаимодействие
 - c. Обмен информацией между приложениями
- 25. Является ли logcat полным аналогом gdb?**
- a. Да
 - b. logcat — полный аналог gdbserver
 - c. Нет
- 26. Из-за чего glibc была заменена на Bionic?**
- a. Смена бренда
 - b. Несовместимость glibc со свободными лицензиями
 - c. glibc невозможно скомпилировать для ARM
 - d. Уменьшение размеров объектного кода
- 27. Какой байт-код используется в Dalvik?**
- a. Обычный
 - b. Исключительно свой собственный
 - c. Свой собственный, но обычный может быть в него сконвертирован
- 28. Из чего состоит Java API в Android?**
- a. Часть стандартных библиотек Java
 - b. Стандартные библиотеки Java и пакет javax.
 - c. Набор стандартных, популярных и собственных пакетов
 - d. Исключительно собственные библиотеки Android
- 29. Какую функциональность обеспечивает пакет android.view?**
- a. Реализация GUI
 - b. Работа с камерой
 - c. Темы рабочего стола
- 30. Какой способ беспроводной связи есть в пакете android.net?**
- a. BlueTooth
 - b. WiFi
 - c. NFC
 - d. DRM

Тема 1.2 Создание и тестирование модулей для мобильных приложений

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется *в электронном виде с помощью Onlinetestpad.com*

Инструкция: на выполнение теста отводится 30 минут, внимательно прочитайте вопрос, выберите один вариант ответа и заносится в бланк ответов

Контрольные вопросы для проведения тестирования

Вариант 1

1. Какое из перечисленных преимуществ создания модулей для мобильных приложений является наиболее значимым?
 - a) Снижение затрат на разработку
 - b) Увеличение производительности
 - c) Удобство в использовании
 - d) Большая гибкость при обновлении
2. Какой инструмент необходим для создания модулей для мобильных приложений?
 - a) Интегрированная среда разработки
 - b) Программа для создания анимаций
 - c) Компилятор языка C++
 - d) База данных
3. Какой из следующих методов тестирования модулей является наиболее надежным?
 - a) Тестирование на реальном устройстве
 - b) Тестирование на эмуляторе
 - c) Тестирование в симуляторе
 - d) Ручное тестирование
4. Какой из перечисленных типов модулей для мобильных приложений отвечает за взаимодействие с веб-сервисами?
 - a) Модуль авторизации
 - b) Модуль работы с графикой
 - c) Модуль управления данными
 - d) Модуль работы с сетью
5. Какие из перечисленных задач относятся к тестированию модулей для мобильных приложений?
 - a) Проверка корректности работы алгоритмов
 - b) Оценка удобства интерфейса
 - c) Тестирование совместимости с различными устройствами
 - d) Разработка дизайна
6. Какой метод тестирования модулей для мобильных приложений позволяет выявить проблемы с памятью?
 - a) Тестирование на производительность
 - b) Тестирование на утечки памяти
 - c) Тестирование на соответствие требованиям
 - d) Тестирование на безопасность
7. Какая функция отвечает за обработку ошибок в модуле для мобильного приложения?
 - a) try-catch

- b) for-in
 - c) if-else
 - d) switch-case
8. Какой из перечисленных форматов файлов наиболее подходит для хранения данных, используемых в модуле для мобильного приложения?
- a) XML
 - b) MP3
 - c) JPG
 - d) AVI
9. Какой из перечисленных подходов к созданию модулей для мобильных приложений наиболее распространен?
- a) Ручное создание модулей
 - b) Использование сторонних библиотек
 - c) Автоматическое создание модулей
 - d) Использование готовых шаблонов
10. Какой вид модуля для мобильного приложения отвечает за обработку взаимодействия пользователя с приложением?
- a) Модуль работы с данными
 - b) Модуль управления событиями
 - c) Модуль работы с графикой
 - d) Модуль работы с сетью
11. Какой инструмент используется для автоматического тестирования модулей для мобильных приложений?
- a) Программное обеспечение для тестирования
 - b) Интегрированная среда разработки
 - c) База данных
 - d) Программа для создания анимаций
12. Какой из перечисленных форматов файлов используется для хранения и передачи данных в мобильных приложениях?
- a) JSON
 - b) BMP
 - c) GIF
 - d) WAV
13. Какой вид тестирования модулей для мобильных приложений позволяет выявить ошибки в работе приложения при нестандартных условиях?
- a) Функциональное тестирование
 - b) Тестирование на производительность
 - c) Тестирование на надежность
 - d) Тестирование на стабильность
14. Какой из перечисленных методов тестирования модулей для мобильных приложений используется для проверки работы на разных версиях операционной системы?
- a) Тестирование на реальных устройствах
 - b) Тестирование на эмуляторе
 - c) Тестирование в симуляторе
 - d) Тестирование на совместимость

15. Какой инструмент позволяет упростить тестирование модулей для мобильных приложений, позволяя быстро находить и исправлять ошибки?
- a) Интегрированная среда разработки
 - b) Программа для автоматизации тестирования
 - c) База данных
 - d) Программа для создания анимаций

Вариант 2

1. Набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения называется:

- a. Android SDK
- b. JDK
- c. плагин ADT
- d. Android NDK

2. С какой целью был создан Open Handset Alliance?

- a. писать историю развития ОС Android
- b. продавать смартфоны под управлением Android
- c. рекламировать смартфоны под управлением Android
- d. разрабатывать открытые стандарты для мобильных устройств

3. С какой целью инструмент Intel* Graphics Performance Analyzers (Intel* GPA) System Analyzer используется в среде разработки Intel* Beacon Mountain?

- a. позволить разработчикам оптимизировать загрузенность системы при использовании процедур OpenGL
- b. для ускорения работы эмулятора в среде разработки
- c. для оптимизированной обработки данных и изображений
- d. позволить разработчикам эффективно распараллелить C++ мобильные приложения

4. Библиотеки, реализованные на базе PacketVideo OpenCORE:

- a. Media Framework
- b. SQLite
- c. FreeType
- d. 3D библиотеки

a. Какой движок баз данных используется в ОС Android?

- a. InnoDB
- b. DBM
- c. MyISAM
- d. SQLite

5. С какой целью инструмент Intel* Integrated Performance Primitives (Intel* IPP) используется в среде разработки Intel* Beacon Mountain?

- a. для оптимизированной обработки данных и изображений
- b. позволить разработчикам оптимизировать загрузенность системы при использовании процедур OpenGL
- c. для ускорения работы эмулятора в среде разработки

- d. позволить разработчикам эффективно распараллелить C++ мобильные приложения
- 6. **Intel XDK поддерживает разработку под:**
 - a. JavaFX Mobile
 - b. Apple iOS, BlackBerry OS
 - c. MtkOS, Symbian OS, Microsoft Windows 8
 - d. Android, Apple iOS, Microsoft Windows 8, Tizen
- 7. **Каждый приемник широковещательных сообщений является наследником класса**
 - a. ViewReceiver
 - b. IntentReceiver
 - c. ContentProvider
 - d. BroadcastReceiver
- 8. **Какой класс является основным строительным блоком для компонентов пользовательского интерфейса (UI), определяет прямоугольную область экрана и отвечает за прорисовку и обработку событий?**
 - a. GUI 27
 - b. View
 - c. UIComponent
 - d. Widget
- 9. **Какой слушатель используется для отслеживания события касания экрана устройства?**
 - a. OnPressListener
 - b. onTouchListener
 - c. OnClickListener
 - d. OnInputListener
- 10. **В какой папке необходимо разместить XML файлы, которые определяют все меню приложения?**
 - a. res/value
 - b. res/items
 - c. res/layout
 - d. res/menu
- 11. **Фоновые приложения ...**
 - a. после настройки не предполагают взаимодействия с пользователем, большую часть времени находятся и работают в скрытом состоянии
 - b. выполняют свои функции и когда видимы на экране, и когда скрыты другими приложениями
 - c. небольшие приложения, отображаемые в виде графического объекта на рабочем столе
 - d. большую часть времени работают в фоновом режиме, однако допускают взаимодействие с пользователем и после настройки
- 12. **Полный иерархический список обязательных файлов и папок проекта можно увидеть на вкладке ...**
 - a. Package Explorer
 - b. Internet Explorer
 - c. Navigator

d. Project Explorer

13. Какой компонент управляет распределенным множеством данных приложения?

- a. сервис (Service)
- b. активность (Activity)
- c. приемник широковещательных сообщений (Broadcast Receiver)
- d. контент-провайдер (Content Provider)

14. Какой язык разметки используется для описания иерархии компонентов графического пользовательского интерфейса Android-приложения?

- a. html
- b. xml
- c. gml
- d. xhtml

15. Выберите верную последовательность действий, необходимых для создания в приложении контент-провайдера

- a. Создание класса наследника от класса ContentProvider; Определение способа организации данных; Заполнение контент-провайдера данными
- b. Проектирование способа хранения данных; Определение способа организации данных;
- c. Создание класса наследника от класса ContentProvider; Заполнение контент-провайдера данными; Определение способа работы с данными
- d. Проектирование способа хранения данных; Создание класса-наследника от класса ContentProvider; Определение строки авторизации провайдера, URI для его строк и имен столбцов

16. Выберите верные утверждения относительно объекта-намерения (Intent).

- a. представляет собой структуру данных, содержащую описание операции, которая должна быть выполнена, и обычно используется для запуска активности или сервиса
- b. используется для передачи сообщений пользователю
- c. используется для получения инструкций от пользователя
- d. используются для передачи сообщений между основными компонентами приложений

17. Расположение элементов мобильного приложения:

- a. полезно для передачи иерархии
- b. влияет на удобство использования
- c. полезно для создания пространственных отношений между объектами на экране и объектами реального мира
- d. все варианты ответа верны

18. Какие элементы управления применяются для действий по настройке?

- a. командные элементы управления
- b. элементы выбора
- c. элементы ввода
- d. элементы отображения

19. Примерами комбо-элементов не являются:

- a. комбо-список
- b. все вышеперечисленное

- c. комбо-кнопка
 - d. комбо-поле
20. **Дизайн или проектирование интерфейса для графических дизайнеров:**
- a. все варианты ответа верны
 - b. прозрачность и понятность информации
 - c. тон, стиль, композиция, которые являются атрибутами бренда
 - d. передача информации о поведении посредством ожидаемого назначения
21. **Более крупные элементы:**
- a. привлекают больше внимания
 - b. все варианты ответа верны
 - c. размер не влияет на уровень внимания
 - d. привлекают меньше внимания
22. **К традиционным типографическим инструментам не относят**
- a. масштаб
 - b. цвет
 - c. разреженность
 - d. выравнивание по сетке
23. **К элементам ввода относят:**
- a. ограничивающие элементы ввода
 - b. ползунки
 - c. счетчики
 - d. все вышеперечисленное
24. **24) Выделяют следующие категории плотности экрана для Android-устройств:**
- a. HDPI, XHDPI, XXHDPI, и XXXHDPI
 - b. правильный вариант ответа отсутствует
 - c. LDPI, MDPI, HDPI, XHDPI, XXHDPI, и XXXHDPI
 - d. LDPI, MDPI, HDPI
25. **Следующие утверждения не верны:**
- a. не используйте интерфейсные элементы
 - b. картинки работают быстрее, чем слова
 - c. на любом шаге должна быть возможность вернуться назад
 - d. если объекты похожи, они должны выполнять сходные действия
26. **Следующие утверждения верны:**
- a. текстура бесполезна для передачи различий или привлечения внимания
 - b. восприятие направления затруднено при больших размерах объектов
 - c. все варианты ответа верны
 - d. люди легко воспринимают контрастность
27. **Основные вкладки (FixedTabs) удобны при отображении**
- a. от четырех вкладок
 - b. двух вкладок
 - c. трех и более вкладок
 - d. трех и менее вкладок
28. **Диалоговое окно, содержащее линейку процесса выполнения какого-то действия — это**
- a. DatePickerDialog
 - b. AlertDialog

- c. ProgressDialog
- d. DialogFragment

29. Уведомления стоит использовать, когда

- a. сообщение не требует ответа пользователя, но важно для продолжения его работы
- b. сообщение является важным и требует немедленного прочтения и ответа
- c. сообщение является важным, однако требует немедленного прочтения, но не ответа
- d. сообщение является важным, однако не требует немедленного прочтения и ответа

30. Какой метод запускает новую активность?

- a. startActivity()
- b. beginActivity()
- c. intentActivity()
- d. newActivity()

31. ProgressDialog это:

- a. контейнер для создания собственных диалоговых окон
- b. диалоговое окно с предопределенным интерфейсом, позволяющее выбрать дату или время
- c. диалоговое окно, содержащее линейку процесса выполнения какого-то действия
- d. диалоговое окно, которое может содержать заголовок, до трех кнопок, список выбираемых значений или настраиваемое содержимое

32. AlertDialog это:

- a. контейнер для создания собственных диалоговых окон
- b. диалоговое окно, содержащее линейку процесса выполнения какого-то действия
- c. диалоговое окно, которое может содержать заголовок, до трех кнопок, список выбираемых значений или настраиваемое содержимое
- d. диалоговое окно с предопределенным интерфейсом, позволяющее выбрать дату или время

33. Что необходимо сделать при добавлении в проект новой активности?

- a. скачать и установить специальный инструмент MultiActivity SDK
- b. прописать в манифесте информацию о новой активности
- c. создать новый проект
- d. запустить эмулятор

34. Системы позиционирования смартфона могут включать

- a. все перечисленное
- b. систему GPS
- c. систему ГЛОНАСС
- d. сигналы WiFi и Bluetooth

35. Какая константа не определена в классе MotionEvent, для обозначения сенсорных событий

- a. ACTION_DOWN 30
- b. ACTION_UP
- c. ACTION_CLICK
- d. ACTION_MOVE

36. С какой целью используется метод release() в классах MediaPlayer и MediaRecorder?

- a. конец жизненного цикла объекта и освобождение ресурсов

- b. перевод объекта в ожидающее состояние
 - c. обновление объекта и запуск его работы
 - d. создание объекта и запуск его работы
37. **К датчикам окружающей среды, встроенным в мобильное устройство относят**
- a. датчики вектора вращения
 - b. датчики освещенности
 - c. акселерометры
 - d. гироскопы
38. **Библиотека Universal Image Loader for Android позволяет:**
- a. парсить HTML-страницы
 - b. строить графики и диаграммы
 - c. загружать, кешировать и отображать изображения
 - d. использовать анимацию, доступную только с версии 3.x, на более ранних вариантах платформы Android
39. **Facebook SDK for Android — это библиотека, позволяющая:**
- a. получать доступ к информации любого пользователя
 - b. отправлять рекламные сообщения от имени пользователя
 - c. писать сообщения на стену, читать и менять статусы, смотреть ленту друзей
 - d. парсить страницы пользователей
40. **Что из перечисленного не относится к правилам безопасности при подключении библиотек?**
- a. с осторожностью использовать библиотеки из сомнительных источников
 - b. ознакомиться с форумами и сайтами, где могут обсуждаться библиотеки
 - c. лично познакомиться с разработчиками библиотеки
 - d. использовать скомпрометированные библиотеки
41. **Библиотека MapNavigator предназначена для:**
- a. работы с любыми картами
 - b. работы с Яндекс.Картами
 - c. морской навигации
 - d. работы с картами Google Maps
42. **Библиотека jsoup не позволяет:**
- a. находить и извлекать данные, используя DOM и селекторы CSS
 - b. манипулировать HTML-элементами, атрибутами и текстом
 - c. писать сообщения на стену, читать и менять статусы, смотреть ленту друзей
 - d. принимать в качестве параметра URL, файл или строку
43. **При настройке обратной совместимости необходимо добавить в файл манифеста следующую информацию:**
- a. только минимальную версию Android SDK
 - b. минимальную и основную (целевую) версии Android SDK
 - c. информацию о подключенной библиотеке
 - d. только основную (целевую) версию Android SDK
44. **Какая библиотека предназначена для упрощения загрузки изображений?**
- a. Yandex.Metrica for Apps;
 - b. Universal Image Loader for Android
 - c. ActionBarSherlock
 - d. NineOldAndroids

45. **Библиотеки совместимости предназначены для**
- сбора статистики
 - рисования графиков
 - использования возможностей, появившиеся в какой-то версии ОС Android, на более ранних версиях платформы
 - подключения нестандартных элементов управления
46. **Какая библиотека предназначена для использования анимации?**
- Universal Image Loader for Android
 - NineOldAndroids
 - Yandex.Metrica for Apps
 - ActionBarSherlock
47. **Для чего служит папка res/anim/ проекта?**
- в этой папке находятся файлы, содержащие набор картинок, предназначенных для кадровой анимации
 - в этой папке находятся файлы, содержащие анимированные ролики для воспроизведения в приложении
 - в этой папке находятся XML файлы, задающие реализацию анимации свойств
 - в этой папке находятся XML файлы, задающие последовательность инструкций анимации преобразований
48. **В какой файл обязательно добавляется информация при создании нового Activity в приложении?**
- AndroidManifest.xml
 - main.java
 - layout.xml
 - activity.xml
49. **Какой метод жизненного цикла активности вызывается системой непосредственно перед появлением активности на экране?**
- onVisible()
 - onOpen()
 - onResume()
 - onCreate()
50. **С какой целью используется метод SurfaceHolder.lockCanvas()?**
- блокировка Canvas для перерисовки
 - игнорирование дальнейшего взаимодействия с Canvas
 - сокрытие Canvas
 - блокировка Canvas от сворачивания
51. **Может ли мобильное приложение получить доступ к базе данных, созданной в другом приложении?**
- не может ни при каких обстоятельствах
 - может, но только с помощью контент-провайдеров
 - право на доступ открывает приложение-хозяин базы данных
 - может обращаться напрямую
52. **С помощью какого метода можно запретить смену ориентации устройства, при запущенном приложении?**
- setRequestedOrientation
 - setChangeOrientation

- c. disableChangeOrientation
- d. setOrientation

53. Какой из датчиков не используется для определения положения смартфона в пространстве?

- a. акселерометр
- b. gps
- c. гироскоп
- d. магнитометр

54. К новым возможностям HTML5 относят:

- a. возможность добавления аудио и видео без использования вспомогательных средств
- b. возможность рисования на холсте
- c. возможность прямого доступа к оперативной памяти
- d. форматирование данных в режиме таблицы

55. Возможен ли перенос приложений iOS* в среду HTML5:

- a. нет, прямой перенос приложений невозможен
- b. да, используя средства Intel XDK
- c. да, используя только сторонние средства
- d. да, только для iPhone, используя средства Intel XDK

56. Следующие утверждения верны:

- a. JavaScript не позволяет подключать другие внешние библиотеки, написанные на других языках
- b. приложения html5 исполняются быстрее и требуют меньше ресурсов, чем «нативные»
- c. среда Intel XDK не работает с мультисенсорностью
- d. приложения html5 исполняются медленнее и требуют больших ресурсов, чем «нативные»

57. Разрабатывать приложения в среде Intel XDK можно:

- a. пользоваться заготовленными примерами
- b. все варианты ответа верны
- c. «с нуля», прописывая все элементы
- d. использовать встроенный «дизайнер элементов» для отрисовки элементов

58. JavaScript не позволяет:

- a. получать прямой доступ к памяти
- b. работать с реестром
- c. работать с картами
- d. одновременно использовать несколько подключаемых библиотек

59. В среде Intel XDK можно разрабатывать приложения для следующих платформ:

- a. Android
- b. все варианты ответа верны
- c. Apple iOS
- d. Tizen

60. В заготовке любого приложения, разрабатываемого в среде Intel XDK прописана:

- a. все варианты ответа верны

- b. скрывание окна заставки Intel XDK
- c. настройка размеров приложения под размеры устройства
- d. фиксация размеров приложения (запрет «скроллинга»)

4 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

МДК 01.03 Разработка мобильных приложений

Изучение *МДК 01.03 Разработка мобильных приложений* реализуется в течение 1 семестра.

Формами контроля промежуточной аттестации *МДК 01.03 Разработка мобильных приложений* является экзамен в 5 семестре.

Основой для определения оценки при проведении промежуточных аттестаций служит объём и уровень усвоения обучающимися материала, предусмотренного рабочей программой профессионального модуля *ПМ.01 Разработка модулей программного обеспечения для компьютерных систем* в части *МДК 01.03 Разработка мобильных приложений*.

Экзамен

Экзамен проводится по завершению курса изучения МДК в форме собеседования по билетам, содержащим два теоретических вопроса и практическое задание.

К сдаче экзамена допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации за семестр оценки не ниже «удовлетворительно».

Вопросы для подготовки к экзамену

1. Архитектура платформы Android. Уровень ядра. Уровень библиотек.
2. Архитектура платформы Android. Dalvik Virtual Machine.
3. Архитектура платформы Android. Уровень каркаса приложений. Уровень приложений.
4. Среда разработки для Android. Eclipse IDE. Плагин ADT. Android Virtual Device.
5. Android SDK. Версии SDK и Android API Level.
6. Структура проекта Android-приложения в Eclipse. Каталоги ресурсов. Файл R.java.
7. Графический интерфейс пользователя в Android-приложениях. XMLразметка интерфейса.
8. Архитектура платформы Android.
9. XML-разметка интерфейса пользователя.
10. XAML-разметка интерфейса пользователя.
11. Базовые элементы управления.
12. Ресурсы в Android-приложениях.
13. Ресурсы в Windows Phone-приложениях.
14. Активности и интенды.
15. Обработка пользовательского ввода. Касания, ввод текста.
16. Типы компоновок графического интерфейса. FrameLayout, LinearLayout, TableLayout, RelativeLayout.
17. Базовые элементы управления. TextView. EditText. Тип ввода текста. Параметры отображения клавиатуры. ImageView.

18. Диалоговые окна. AlertDialog. ProgressDialog. DatePickerDialog. TimePickerDialog. Создание пользовательских диалоговых окон.
19. Многопоточные приложения в Android и Windows Phone. Использование системных таймеров и системного времени.
20. Процессы в Android. Объекты Activity. Состояния Activity.
21. Использование объектов Intent. Intent-фильтры.
22. Использование ресурсов. Ссылки на ресурсы. Загрузка простых типов из ресурсов. Загрузка файлов произвольного типа.
23. Файловая система Android. Чтение и запись файлов.
24. Адаптеры данных. Отображение данных в компонентах ListView, GridView, AutoCompleteTextView, MultiAutoCompleteTextView.
25. Пользовательские настройки. Использование SharedPreferences. Виды настроек.
26. Работа с графикой. Drawable и Canvas.
27. Работа с анимацией. Tween Animation и Frame Animation. Описание анимации в XML и в коде программы.
28. Службы в Android. Компонент Service.
29. Датчики мобильных устройств. Управление датчиками в приложении. Виды датчиков и особенности их использования.
30. Программный доступ к дисплею устройства. Менеджер окон. Параметры дисплея.

Перечень заданий для оценки усвоенных умений

Задание 1. Скачать Android SDK + Eclipse (Eclipse ADT Bundle) <http://developer.android.com/sdk/index.html>, а также последнюю версию Java (JDK) <http://www.oracle.com/technetwork/java/index-jsp-138363.html> (или просто набрать в yandex "JDK" и пройти по ссылке на oracle.com). Установить всё это. Создать новый проект, зайти в Android SDK и скачать какую-нибудь версию Android SDK (например, lollipop - 5.0.1), выделив все инструменты для работы с ней; сконфигурировать эмулятор (желательно эмулировать своё собственное мобильное устройство, включая версию андроид на нём в качестве Target SDK). Изменить TextView с надписью "Hello world" на какую-нибудь другую надпись по желанию. Запустить проект на эмуляторе и убедиться, что всё работает.

Задание 2. Создать новый проект, написать программу, которая выводит в элемент TextView надпись, введенную пользователем в текстовом поле EditText после нажатия на кнопку Button. Помимо этого в Activity должен быть TextView с ФИО студента и группой. Запустить на эмуляторе и убедиться, что всё работает.

Задание 3. Создать приложение, которое состоит из нескольких activities. Первое activity содержит элемент TextView с названием или номером activity, текстовое поле EditText для ввода какой-то информации, кнопку Button с названием "Next" или "Перейти на 2 activity/экран/окно" или просто "2". Помимо этого в 1 activity должен быть TextView с ФИО студента и группой. После нажатия на эту кнопку происходит переход на второе activity, где содержится TextView с названием или номером activity, TextView с надписью что-то вроде "В первом окне вы напечатали:" и под ним - ещё один TextView с содержимым EditText с первого activity, и, разумеется, кнопка "1" или "Вернуться на 1 экран" или "Вернуться к вводу текста", нажав на которую пользователь может перейти обратно к 1 activity. Запустить на эмуляторе и убедиться, что всё работает.

Задание 4. В новом проекте написать приложение, работающее с разными темами/стилями. Сначала создать свой стиль и применить его к какому-нибудь интерфейсному

элементу, затем - свою тему, которая применяется ко всем интерфейсным элементам. Приложение при этом должно выглядеть нестандартно, запустить на эмуляторе и убедиться, что всё работает. При возникновении ошибок открыть лог (CatLog) внизу, найти первую красную надпись и породить свою тему от той, которая требуется в этом красном сообщении.

Задание 5. Создать пользовательский (свой) список. Например, получить доступ в приложении к контактам (Permissions-закладка в AndroidManifest.xml) и скопировать контакты телефона в свой список, который отобразить после запуска приложения. Или создать свой список в виде твиттера (картинка+текст), элементы которого просто статически задать в массиве (как и картинки).

Задание 6. Создать приложение, содержащее анимированные интерфейсные элементы (например, увеличивающиеся при клике на них кнопки, вращающиеся TextView и т.д.).

Задание 7. Создать приложение, отображающее после запуска карты Google или какие-нибудь другие карты.

Задание 8. Создать собственный виджет с настройками. Например, виджет, который открывает какой-то сайт, адрес которого можно поменять в настройках.

Задание 9. Создать приложение, использующее опциональное меню (меню настроек) и контекстное меню для какого-нибудь интерфейсного элемента. Естественно, выбор пунктов меню должен что-то менять в интерфейсных элементах или их отображении! Например, очистить поле ввода через контекстное меню, или отобразить невидимые интерфейсные элементы через установку галок в опциональном меню.

Задание 10. Создать приложение, отображающее после некоторых действий (нажатия на кнопку, например, или проверки корректности ввода текста в EditText) диалоговое окно, свидетельствующее об ошибке/информирующее/предупреждающее пользователя о чём-то.

Задание 11. Создать приложение, помещающее по нажатию на кнопку какое-то сообщение со звуком в панель уведомлений/статус-панель па эмуляторе.

Задание 12. Создать приложение, по нажатию кнопки в котором проигрывается какой-то звук.

Задание 13. Создать приложение, при запуске которого проигрывается какое-то видео.

Задание 14. Создать приложение, при запуске которого активируется фотокамера телефона, производится снимок, и этот снимок помещается в Image View интерфейса приложения.

Задание 15. Создать приложение, работающее с SharedPreferences и сохраняющее настройки, а также работающее с БД SQLite - заполняющее БД по нажатию кнопки 1 с помощью EditText, и выводящее вес записи этой БД в какой-нибудь интерфейсный элемент ниже с помощью кнопки 2 (в виде списка, datagrid или просто правильно настроенного Text View).

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного учреждения
высшего образования
«Пермский национальный исследовательский политехнический университет»

ОЦЕНОЧНЫЕ СРЕДСТВА

МЕЖДИСЦИПЛИНАРНОГО КУРСА

МДК. 01.04 Системное программирование

основной профессиональной образовательной программы
подготовки специалистов среднего звена
по специальности СПО 09.02.07 Информационные системы и программирование
(базовая подготовка)

Лысьва, 2023

ОБЩИЕ ПОЛОЖЕНИЯ

Контрольно – измерительные материалы по результатам изучения междисциплинарного курса **МДК. 01.04 Системное программирование** ориентированы на проверку степени достижения требований к минимуму содержания и уровню подготовки обучающихся в соответствии с ФГОС и является основополагающим документом для организации контроля знаний, умений, практического опыта обучающихся в учебном процессе.

Результатом освоения междисциплинарного курса **МДК. 01.04 Системное программирование** является:

Код и наименование профессиональных и общих компетенций, личностных результатов, формируемых в рамках ПМ	Основные показатели оценки результата
ПК 1.2 <i>Разрабатывать программные модули в соответствии с техническим заданием</i>	<i>Уметь:</i> – осуществлять разработку кода программного модуля на языках низкого и высокого уровней – создавать программу по разработанному алгоритму как отдельный модуль <i>Знать:</i> – основные этапы разработки программного обеспечения – основные принципы технологии структурного и объектно-ориентированного программирования
ПК 1.3 <i>Выполнять отладку программных модулей с использованием специализированных программных средств</i>	<i>Уметь:</i> — выполнять отладку и тестирование программы на уровне модуля <i>Знать:</i> — основные принципы отладки и тестирования программных продуктов
ЛР 5	Демонстрирующий умение эффективно взаимодействовать в команде, вести диалог, в том числе с использованием средств коммуникации
ЛР 6	Демонстрирующий навыки анализа и интерпретации информации из различных источников с учетом нормативно-правовых норм
ЛР 7	Демонстрирующий готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни; сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности.
ЛР 8	Пользоваться профессиональной документацией на государственном и иностранном языках.
ЛР 9	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих

	ценностей, применять стандарты антикоррупционного поведения
<i>ЛР 10</i>	Использовать знания по финансовой грамотности, планировать предпринимательскую деятельность в профессиональной сфере.
<i>ЛР 11</i>	Планировать и реализовывать собственное профессиональное и личностное развитие в условиях развития информационных технологий, применяемых в различных отраслях народного хозяйства.
<i>ЛР 12</i>	Активно применяющий полученные знания на практике
<i>ЛР 13</i>	Способный анализировать производственную ситуацию, быстро принимать решения
<i>ЛР 14</i>	Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
<i>ЛР 17</i>	Проявлять доброжелательность к окружающим, деликатность, чувство такта и готовность оказать услугу каждому кто в ней нуждается.

1 МЕТОДЫ И ФОРМЫ КОНТРОЛЯ ОЦЕНИВАНИЯ ЭЛЕМЕНТОВ МЕЖДИСЦИПЛИНАРНОГО КУРСА

МДК. 01.04 Системное программирование

1 Для текущего и рубежного контроля освоения дисциплинарных компетенций используются следующие методы:

- Устный опрос
- Наблюдение и оценка результатов практических занятий
- Наблюдение и оценка результатов лабораторных занятий
- Экспертная оценка результатов самостоятельной работы
- Экспертная оценка по результатам наблюдения за деятельностью обучающегося в

процессе освоения ПМ

2 Формой контроля промежуточной аттестации междисциплинарного курса являются: дифференцированный зачет (6 семестр), экзамен (7 семестр), которые проводятся в сроки, установленные учебным планом и определяемые календарным учебным графиком образовательного процесса.

Таблица 1 – Методы и формы контроля и оценивания элементов междисциплинарного курса
МДК. 01.04 Системное программирование

Элемент учебной дисциплины	Методы и формы контроля и оценивания		
	Текущий контроль	Рубежный контроль	Промежуточная аттестация
Раздел 1 Программная модель процессора			
Тема 1.1 <i>Программная модель процессора</i>	Устный опрос Наблюдение и оценка результатов практических занятий Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	<i>Тестирование</i>	<i>Дифференцированный зачет (6 семестр)</i>
Раздел 2 Программирование на языке низкого уровня			

Тема 2.1 Программирование на языке низкого уровня	Устный опрос Наблюдение и оценка результатов лабораторных занятий Экспертная оценка результатов самостоятельной работы Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ	<i>Тестирование</i>	<i>Экзамен(7семестр)</i>
Форма контроля			<i>Дифференцированный зачет Экзамен</i>

Текущий контроль усвоения материала

Текущий контроль усвоения материала проводится в форме устного опроса студентов по темам дисциплины.

Наблюдение и оценка результатов практических занятий

Типовые темы практических занятий приведены в РПД. Комплект заданий на практические занятия приведены в МУ по ПЗ по учебной дисциплине.

Защита отчетов по практическим занятиям проводится индивидуально каждым обучающимся в форме собеседования.

Наблюдение и оценка результатов лабораторных занятий

Типовые темы лабораторных работ приведены в РПД. Комплект заданий на лабораторные занятия приведены в МУ по ЛР по учебной дисциплине.

Защита отчетов по лабораторным занятиям проводится индивидуально каждым обучающимся в форме собеседования.

Экспертная оценка результатов самостоятельной работы

Задания для самостоятельной работы приведены в МУ по СРС по учебной дисциплине.

Качественная оценка определения научного кругозора, степенью овладения методами теоретического исследования и развития самостоятельности мышления студента.

Способом проверки качества организации самостоятельной работы студентов является контроль:

- корректирующий (может осуществляться во время индивидуальных консультаций по поводу выполнения формы самостоятельной работы);
- констатирующий (по результатам выполнения специальных форм самостоятельной работы);
- самоконтроль (осуществляется самим студентом);
- текущий (в ходе выполнения различных форм самостоятельной работы, установленных рабочей программой);
- промежуточный (оценка результата обучения как итога выполнения студентом всех форм самостоятельной работы).

Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения учебной дисциплины

Осуществляется как наблюдение за процессом деятельности обучающегося в режиме реального времени. Является качественной оценкой освоения учебной дисциплины, учитываемой при промежуточной аттестации.

Рубежный контроль

Рубежный контроль для комплексного оценивания усвоенных знаний, усвоенных умений проводится в форме, тестирования (после изучения разделов учебной дисциплины).

2 РЕЗУЛЬТАТЫ ОСВОЕНИЯ ЭЛЕМЕНТОВ МЕЖДИСЦИПЛИНАРНОГО КУРСА, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ ПРИ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

В результате промежуточной аттестации междисциплинарного курса осуществляется комплексная проверка следующих умений, знаний:

Результаты обучения (освоенные умения, усвоенные знания)	Показатели оценки результатов
Уметь:	
осуществлять разработку кода программного модуля на языках низкого и высокого уровней	Умеет осуществлять разработку кода программного модуля на языках низкого уровня
создавать программу по разработанному алгоритму как отдельный модуль	Умеет создавать программу по разработанному алгоритму как отдельный модуль
выполнять отладку и тестирование программы на уровне модуля	Умеет выполнять отладку и тестирование программы на уровне модуля
Знать:	
основные этапы разработки программного обеспечения	Знает основные этапы разработки программного обеспечения
основные принципы технологии структурного и объектно-ориентированного программирования	Знает основные принципы технологии структурного программирования
основные принципы отладки и тестирования программных продуктов	Знает основные принципы отладки и тестирования программных продуктов

3 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ ЗНАНИЙ И УМЕНИЙ

МДК 01.04 Системное программирование

Задания для оценки освоения *Раздела 1 Программная модель процессора* темы 1.1 Программная модель процессора

Обучающийся должен

знать:

- основные этапы разработки программного обеспечения
- основные принципы технологии структурного и объектно-ориентированного программирования
- основные принципы отладки и тестирования программных продуктов

уметь:

- осуществлять разработку кода программного модуля на языках низкого и высокого уровней
- создавать программу по разработанному алгоритму как отдельный модуль
- выполнять отладку и тестирование программы на уровне модуля

Типовые вопросы для устного опроса

1. Как вы понимаете термин "программная модель процессора"?
2. Какие задачи может выполнять процессор в программной модели?
3. Каковы основные компоненты программной модели процессора?
4. Как программа взаимодействует с программной моделью процессора?
5. Каковы преимущества использования программной модели процессора в разработке программного обеспечения?
6. Какие языки программирования подходят для работы с программной моделью процессора?
7. Какие типы инструкций могут быть выполнены процессором в программной модели?
8. Какие типы данных могут обрабатываться в программной модели процессора?
9. Какова роль программной модели процессора в симуляции работы процессора?
10. Какие проблемы могут возникать при работе с программной моделью процессора и как их можно решить?

Типовой тест

Тема 1.1 Программная модель процессора

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется **в электронном виде с помощью [Onlinetestpad.com](https://onlinetestpad.com)**

Инструкция: на выполнение теста отводится 60 минут, внимательно прочитайте вопрос, выберите один вариант ответа и заносится в бланк ответов

Контрольные вопросы для проведения тестирования

1. Что такое программная модель процессора?
 - а) Способ описания работы процессора в программе

- b) Модель процессора, созданная на основе программного обеспечения
 - c) Алгоритм работы процессора, описанный на языке программирования
 - d) Модель процессора, созданная на основе аппаратного обеспечения
2. Для чего используется программная модель процессора?
- a) Для создания программного обеспечения, работающего на процессоре
 - b) Для создания аппаратного обеспечения, совместимого с процессором
 - c) Для симуляции работы процессора в программе
 - d) Для управления работой процессора в операционной системе
3. Как программа взаимодействует с программной моделью процессора?
- a) Программа загружает модель процессора на жесткий диск
 - b) Программа передает инструкции процессору через системную шину
 - c) Программа моделирует работу процессора на уровне аппаратного обеспечения
 - d) Программа обрабатывает данные, полученные от процессора в режиме реального времени
4. Какие языки программирования подходят для работы с программной моделью процессора?
- a) Любые языки программирования
 - b) Только ассемблер
 - c) Языки программирования, поддерживающие работу с бинарными данными
 - d) Языки программирования, поддерживающие работу с модулями ядра операционной системы
5. Какие типы инструкций могут быть выполнены процессором в программной модели?
- a) Только арифметические и логические
 - b) Только инструкции работы с памятью
 - c) Любые инструкции, поддерживаемые процессором
 - d) Только инструкции, реализованные в модели процессора
6. Какие типы данных могут обрабатываться в программной модели процессора?
- a) Только числовые данные
 - b) Только бинарные данные
 - c) Любые данные, поддерживаемые процессором
 - d) Только данные, представленные в формате ASCII
7. Какие режимы работы существуют в программной модели процессора x86?
- a) Реальный режим и защищенный режим
 - b) Виртуальный режим и супервизорный режим
 - c) Защищенный режим и супервизорный режим
 - d) Реальный режим и виртуальный режим
8. Какие элементы аппаратной части процессора имеют различное поведение в реальном и защищенном режимах работы?
- a) Регистры общего назначения и стековый указатель
 - b) Оперативная память и кэш-память
 - c) Адресное пространство и системная шина
 - d) Арифметическо-логическое устройство и устройство управления памятью
9. Какие преимущества имеет использование защищенного режима работы в программной модели процессора x86?
- a) Улучшение безопасности работы системы
 - b) Ускорение работы программного кода

- c) Упрощение отладки и тестирования ПО
 - d) Уменьшение количества требуемых ресурсов для запуска ПО
10. Какие проблемы могут возникнуть при работе в реальном режиме в программной модели процессора x86?
- a) Ограниченность адресного пространства
 - b) Невозможность работы с современными операционными системами
 - c) Недостаток ресурсов для запуска программного кода
 - d) Ошибки в работе аппаратной части процессора
11. Что такое "виртуальный режим" в программной модели процессора x86?
- a) Режим, в котором работает виртуальная машина
 - b) Режим, в котором запущена 16-битная операционная система
 - c) Режим, в котором выполняются 32-битные приложения в 64-битной операционной системе
 - d) Режим, в котором эмулируется работа старых процессоров x86
12. Какие компоненты аппаратной части процессора могут быть эмулированы в виртуальном режиме?
- a) Центральное процессорное устройство и системная шина
 - b) Кэш-память и оперативная память
 - c) Арифметическо-логическое устройство и устройство управления памятью
 - d) Все компоненты аппаратной части могут быть эмулированы

**Задания для оценки освоения *Раздела Программирование на языке низкого уровня*
темы 2.1 Программирование на языке низкого уровня**

Обучающийся должен

знать:

- основные этапы разработки программного обеспечения
- основные принципы технологии структурного и объектно-ориентированного программирования
- основные принципы отладки и тестирования программных продуктов

уметь:

- осуществлять разработку кода программного модуля на языках низкого и высокого уровней
- создавать программу по разработанному алгоритму как отдельный модуль
- выполнять отладку и тестирование программы на уровне модуля

Типовые вопросы для устного опроса

1. Что такое язык низкого уровня, и как он отличается от языков высокого уровня?
2. Какие преимущества и недостатки существуют при программировании на языке низкого уровня?
3. Какие языки программирования относятся к языкам низкого уровня, и какие из них наиболее распространены?
4. Что такое ассемблер, и какие задачи он решает в программировании на языке низкого уровня?

5. Какие инструкции процессора можно использовать при написании программ на ассемблере?
6. Какая роль отводится регистрам процессора в программировании на языке низкого уровня?
7. Какие особенности отладки и тестирования программ на языке низкого уровня?
8. Какие проблемы могут возникнуть при программировании на языке низкого уровня, и как их можно решить?
9. Какие задачи лучше всего решать с помощью языков низкого уровня, а какие лучше решать на языках высокого уровня?
10. Какие навыки необходимы для эффективного программирования на языке низкого уровня, и как их можно развить?

Типовой тест

Условия выполнения

- тест выполняется в аудитории во время аудиторных занятий;
- тест выполняется в электронном виде с помощью *Onlinetestpad.com*

Инструкция: на выполнение теста отводится 30 минут, внимательно прочитайте вопрос, выберите один вариант ответа и занесите в бланк ответов

1. Какой язык программирования относится к языкам низкого уровня?
 - a) C#
 - b) Java
 - c) Python
 - d) Ассемблер
2. Что такое ассемблер?
 - a) Язык программирования высокого уровня
 - b) Язык программирования низкого уровня
 - c) Компилятор
 - d) Интерпретатор
3. Какие инструкции процессора можно использовать при программировании на языке ассемблера?
 - a) Только арифметические
 - b) Только логические
 - c) Арифметические, логические и управляющие
 - d) Только управляющие
4. Какие преимущества имеет программирование на языке низкого уровня?
 - a) Простота и понятность кода
 - b) Быстродействие и возможность управления ресурсами компьютера
 - c) Удобство разработки многопоточных приложений
 - d) Возможность использования объектно-ориентированного подхода
5. Какие недостатки имеет программирование на языке низкого уровня?
 - a) Сложность отладки и тестирования программ
 - b) Невозможность использования ООП
 - c) Медленная скорость выполнения программ
 - d) Невозможность работы с сетью
6. Какие задачи лучше всего решать на языках низкого уровня?

- a) Задачи, связанные с обработкой данных
 - b) Задачи, связанные с разработкой пользовательских интерфейсов
 - c) Задачи, связанные с сетевым взаимодействием
 - d) Задачи, связанные с автоматизацией бизнес-процессов
7. Какие навыки необходимы для эффективного программирования на языке низкого уровня?
- a) Знание алгоритмов и структур данных
 - b) Опыт работы с базами данных
 - c) Навыки работы с сетью
 - d) Умение пользоваться графическими редакторами
8. Какие регистры процессора могут использоваться в программировании на языке низкого уровня?
- a) Только регистры общего назначения
 - b) Только регистры специального назначения
 - c) Регистры общего и специального назначения
 - d) Регистры сопроцессора
9. Как называется операция присваивания значения регистру процессора в языке ассемблера?
- a) MOV
 - b) ADD
 - c) JMP
 - d) XOR
10. Как называется команда перехода к определенной метке в программе на языке ассемблера?
- a) MOV
 - b) ADD
 - c) JMP
 - d) XOR
11. Какие инструкции процессора используются для условных переходов в программировании на языке ассемблера?
- a) MOV
 - b) ADD
 - c) CMP и условные переходы (JZ, JNZ, JB, JNB и т.д.)
 - d) XOR
12. Какой принцип работы языков высокого уровня более эффективен для решения сложных задач?
- a) Использование встроенных библиотек и функций
 - b) Использование объектно-ориентированного подхода
 - c) Использование языков низкого уровня для оптимизации отдельных участков кода
 - d) Никакой из перечисленных
13. Как называется уровень абстракции, который находится между языком машинного кода и языком высокого уровня?
- a) Язык ассемблера
 - b) Язык C
 - c) Язык Java
 - d) Язык Python

14. Какая функция выполняется при вызове оператора "call" в программировании на языке ассемблера?
- a) Вызов подпрограммы
 - b) Определение метки
 - c) Объявление переменной
 - d) Переход на новую строку кода

4 ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

01.04 Системное программирование

Изучение *01.04 Системное программирование* реализуется в течение 2 семестров.

Формами контроля промежуточной аттестации *01.04 Системное программирование* является дифференцированный зачет в 6 семестре, экзамен в 7 семестре.

Основой для определения оценки при проведении промежуточных аттестаций служит объем и уровень усвоения обучающимися материала, предусмотренного рабочей программой профессионального модуля *ПМ.01 Разработка модулей программного обеспечения для компьютерных систем* в части *01.04 Системное программирование*.

Дифференцированный зачет (6 семестр)

Дифференцированный зачет проводится по завершению изучения раздела МДК по окончании семестра в форме теста с учетом результатов текущего контроля.

К сдаче дифференцированного зачета допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации за семестр оценки не ниже «удовлетворительно».

Вопросы для подготовки к дифференцированному зачету

1. Как вы понимаете термин "программная модель процессора"?
2. Какие задачи может выполнять процессор в программной модели?
3. Каковы основные компоненты программной модели процессора?
4. Как программа взаимодействует с программной моделью процессора?
5. Каковы преимущества использования программной модели процессора в разработке программного обеспечения?
6. Какие языки программирования подходят для работы с программной моделью процессора?
7. Какие типы инструкций могут быть выполнены процессором в программной модели?
8. Какие типы данных могут обрабатываться в программной модели процессора?
9. Какова роль программной модели процессора в симуляции работы процессора?
10. Какие проблемы могут возникать при работе с программной моделью процессора и как их можно решить?
11. Какие регистры общего назначения имеет процессор x86?
12. Какой размер данных обрабатывает процессор x86-32, а какой процессор x86-64?
13. Какие флаги устанавливаются при выполнении арифметических операций в процессоре x86?
14. Какой регистр используется для хранения указателя стека в процессоре x86?
15. Какой регистр используется для хранения адреса возврата из подпрограммы в процессоре x86?

16. Какие режимы работы имеет процессор x86?
17. Какие команды процессора x86 используются для работы с памятью?
18. Какие команды процессора x86 используются для работы с регистрами?
19. Какие команды процессора x86 используются для работы со стеком?
20. Какие операции выполняются при выполнении команды "call" в процессоре x86?

Перечень заданий для оценки усвоенных умений

1. Написать программу, которая складывает два числа, хранящиеся в регистрах EAX и EBX, и сохраняет результат в регистре EAX.
2. Написать программу, которая копирует содержимое одного регистра в другой.
3. Написать программу, которая сравнивает два числа, хранящиеся в регистрах EAX и EBX, и устанавливает флаги CF, ZF, SF, PF в соответствии с результатом сравнения.
4. Написать программу, которая вычисляет сумму чисел в массиве, используя инструкцию LOOP для перебора элементов массива.
5. Написать программу, которая вызывает подпрограмму, передавая ей параметры через стек, и получает результат из регистра EAX.
6. Написать программу, которая использует инструкцию INT для вызова прерывания и выводит на экран строку символов.

Экзамен (7 семестр)

Экзамен проводится по завершению курса изучения МДК в форме собеседования по билетам, которые включают два теоретических и одно практическое задание.

К сдаче экзамена допускаются обучающиеся, выполнившие все отчетные работы и получившие по результатам текущей аттестации за семестр оценки не ниже «удовлетворительно».

Вопросы для подготовки к экзамену

Перечень вопросов для оценки усвоенных знаний

1. Что такое язык низкого уровня, и как он отличается от языков высокого уровня?
2. Какие преимущества и недостатки существуют при программировании на языке низкого уровня?
3. Какие языки программирования относятся к языкам низкого уровня, и какие из них наиболее распространены?
4. Что такое ассемблер, и какие задачи он решает в программировании на языке низкого уровня?
5. Какие инструкции процессора можно использовать при написании программ на ассемблере?
6. Какая роль отводится регистрам процессора в программировании на языке низкого уровня?
7. Какие особенности отладки и тестирования программ на языке низкого уровня?
8. Какие проблемы могут возникнуть при программировании на языке низкого уровня, и как их можно решить?
9. Какие задачи лучше всего решать с помощью языков низкого уровня, а какие лучше решать на языках высокого уровня?

10. Какие навыки необходимы для эффективного программирования на языке низкого уровня, и как их можно развить?
11. Определите, что такое языки низкого уровня и почему они важны для разработки программного обеспечения.
12. Объясните принципы работы процессора и регистров общего назначения.
13. Опишите процесс компиляции и сборки программы на языке низкого уровня.
14. Разберитесь различия между стековой и регистровой архитектурой.
15. Разработайте программу на языке низкого уровня для сортировки массива.
16. Объясните, как работают битовые операции на языке низкого уровня.
17. Напишите программу на языке низкого уровня для работы с файлами.
18. Опишите процесс передачи параметров в подпрограммы на языке низкого уровня.
19. Объясните принципы работы памяти и указателей на языке низкого уровня.
20. Разберитесь принцип работы системных вызовов на языке низкого уровня.

Перечень заданий для оценки усвоенных умений

1. Написать программу на языке ассемблера для умножения двух чисел, хранящихся в регистрах EAX и EBX.
2. Написать программу на языке ассемблера для сортировки массива целых чисел методом выбора.
3. Написать программу на языке ассемблера для вычисления факториала числа, используя рекурсию.
4. Написать программу на языке ассемблера для работы с матрицами: сложение, вычитание, умножение.
5. Написать программу на языке ассемблера для работы с символьными строками: копирование, сравнение, поиск подстроки.
6. Написать программу на языке ассемблера для расчета числа π методом Монте-Карло.
7. Написать программу на языке ассемблера для работы с файлами: чтение, запись, поиск определенного байта или символа.
8. Написать программу на языке ассемблера для работы с сетью: отправка и прием пакетов по протоколу TCP/IP.
9. Написать программу на языке ассемблера для работы с драйверами устройств: инициализация устройства, передача данных.
10. Написать программу на языке ассемблера для работы с оперативной памятью: выделение и освобождение блоков памяти, копирование данных в память.

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного учреждения
высшего образования
«Пермский национальный исследовательский политехнический университет»

ОЦЕНОЧНЫЕ СРЕДСТВА

ЭКЗАМЕН ПО МОДУЛЮ

ПМ.01. РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ

основной профессиональной образовательной программы
подготовки специалистов среднего звена
по специальности СПО 09.02.07 Информационные системы и программирование
(базовая подготовка)

Лысьва, 2023

ОБЩИЕ ПОЛОЖЕНИЯ

Целью проведения экзамена по модулю является оценка соответствия достигнутых компетентностных образовательных результатов обучающихся по профессиональному модулю требованиям ФГОС СПО, готовности обучающихся к определенному виду профессиональной деятельности по избранной специальности.

Оценочные средства предназначены для контроля и оценки результатов освоения профессионального модуля *ПМ.01. Разработка модулей программного обеспечения для компьютерных систем*, который проводится в форме **экзамена по модулю**.

Условием допуска к экзамену по модулю является положительная аттестация по МДК, учебной и производственной (по профилю специальности) практикам.

Экзамен по модулю проводится в виде выполнения практических заданий, имитирующих работу в обычных условиях, направленных на оценку готовности обучающихся, завершивших освоение профессионального модуля, к реализации вида профессиональной деятельности.

Итогом экзамена является однозначное решение: «вид профессиональной деятельности освоен / не освоен», профессиональный модуль: зачтено/не зачтено

1 Комплект экзаменационных материалов

В состав комплекта входит задание для экзаменуемого, пакет экзаменатора и оценочная ведомость.

ЗАДАНИЕ ДЛЯ ЭКЗАМЕНУЮЩЕГОСЯ

Оцениваемые компетенции: ПК1.1 ПК1.2 ПК1.3 ПК1.4 ПК1.5 ПК1.6

Инструкция

1 Внимательно прочитайте задание

2 Выполните задания в строго определенной последовательности

3 После выполнения всех заданий доложите результаты аттестационной комиссии

Вы можете воспользоваться: персональный компьютер с установленным комплектом программного обеспечения согласно требованиям рабочей программе профессионального модуля

Максимальное время выполнения задания: 60 минут

Задание:

ПК 1.1, ПК 1.2

1 Написать алгоритм разработки программного модуля для конкретного технического задания с использованием выбранного языка программирования

ПК 1.3

2 Использовать инструменты дизассемблирования для анализа исполняемого кода и выявления ошибок

ПК 1.4

3 Напишите тест-кейсы для функции входа в систему, проверяющие корректность ввода логина и пароля, проверку соответствия введенных данных с базой данных, и проверку перехода на нужную страницу после успешного входа

ПК 1.5

4 Оцените качество работы интерфейса пользователя, проведя функциональное тестирование на различных устройствах и экранах, проверяющее корректность расположения элементов, их доступность и соответствие дизайну

ПК 1.6

5 Разработайте модуль, который использует API, чтобы получать данные о погоде и выводить их на экран устройства

ПАКЕТ ЭКЗАМЕНАТОРА

Инструкция

- 1 Внимательно изучите информационный блок пакета экзаменатора
- 2 Ознакомьтесь с заданиями для экзаменуемых, оцениваемыми компетенциями и показателями оценки
- 3 Изучите инструмент оценивания профессиональных и общих компетенций.
- 4 Оцените работу обучающихся и заполните экзаменационную ведомость согласно предложенным критериям

Количество вариантов заданий (пакетов заданий) для экзаменуемых: *10*.

Максимальное время выполнения задания 60 минут

Максимальное время проведения экзамена 220 минут

Условия выполнения заданий: компьютерный класс, оснащенный автоматизированными рабочими местами обучающихся и преподавателя, с установленным программным обеспечением, согласно требованиям рабочей программы профессионального модуля

Оборудование: персональный компьютер с установленным программным обеспечением, согласно требованиям рабочей программы профессионального модуля

Задания:

ПК 1.1, ПК 1.2

1. Написать алгоритм разработки программного модуля для конкретного технического задания с использованием выбранного языка программирования.
2. Изменить техническое задание и попросить студента отразить это изменение в своем алгоритме разработки.
3. Реализовать разработанный студентом алгоритм на компьютере и проверить работоспособность полученного программного модуля.
4. Дополнить неполное техническое задание таким образом, чтобы можно было разработать программный модуль.
5. Определить все этапы жизненного цикла программного продукта, связанные с разработкой программного модуля.
6. Проанализировать техническое задание и выявить потенциальные проблемы при разработке программного модуля. Затем попросить его предложить способы решения этих проблем.
7. Разработать алгоритм для обработки предложенного набора данных в соответствии с техническим заданием.
8. Разработать тестовые случаи для проверки работоспособности программного модуля.
9. Оценить затраты на разработку программного модуля в соответствии с техническим заданием.
10. Провести анализ алгоритмов разработки программных модулей, используемых в настоящее время, и предложить свой вариант более эффективного алгоритма.
11. Написать модуль с использованием выбранного фреймворка или библиотеки, соответствующей техническому заданию.
12. Реализовать механизмы защиты программного модуля от внешних угроз, таких как атаки на основе ввода данных.

ПК 1.3

1. Изучить специализированные программные средства для отладки и выбрать наиболее подходящее средство для конкретного задания.
2. Использовать инструменты дизассемблирования для анализа исполняемого кода и выявления ошибок.
3. Использовать отладочную печать для вывода дополнительной информации в процессе выполнения программного модуля.
4. Использовать средства трассировки выполнения программного модуля для нахождения места возникновения ошибки.
5. Использовать инструменты статического анализа для выявления потенциальных проблем в программном модуле.
6. Попросить студента написать программный модуль с известной ошибкой и попросить его исправить эту ошибку с помощью инструментов отладки.

7. Попросить студента провести отладку программного модуля с использованием точек останова (breakpoints).
8. Попросить студента провести отладку многопоточных программных модулей с помощью специализированных инструментов отладки.
9. Попросить студента настроить среду отладки для конкретного языка программирования, включая настройку опций компилятора и отладчика.
10. Попросить студента проанализировать журналы отладки и определить причину возникновения ошибки в программном модуле.

ПК 1.4

1. Напишите тест-кейсы для функции входа в систему, проверяющие корректность ввода логина и пароля, проверку соответствия введенных данных с базой данных, и проверку перехода на нужную страницу после успешного входа.
2. Проведите тестирование на соответствие спецификации поискового модуля, проверяющее правильность работы алгоритмов поиска, соответствие результатов ожидаемым, и работу фильтров поиска.
3. Оцените качество работы интерфейса пользователя, проведя функциональное тестирование на различных устройствах и экранах, проверяющее корректность расположения элементов, их доступность и соответствие дизайну.
4. Напишите тест-кейсы для проверки корректной работы модуля загрузки файлов, проверяющие возможность загрузки файлов различных форматов, проверку ограничений на размер и тип загружаемых файлов, и проверку успешной загрузки на сервер.
5. Проведите тестирование на соответствие спецификации модуля работы с базой данных, проверяющее корректность выполнения запросов, соответствие полученных данных ожидаемым, и работу механизма обновления и удаления данных.
6. Напишите тест-кейсы для проверки корректной работы модуля отправки электронной почты, проверяющие правильность ввода адреса получателя, проверку наличия вложений, и проверку успешной отправки и доставки письма.
7. Проведите тестирование на соответствие требованиям производительности, проверяющее скорость работы приложения в различных ситуациях, количество и скорость обработки запросов, и расходы ресурсов.
8. Оцените уровень безопасности приложения, проведя тестирование на уязвимости, проверяющее возможность атак на систему, утечки данных, и работу механизма авторизации и аутентификации.
9. Напишите тест-кейсы для проверки корректной работы модуля генерации отчетов, проверяющие правильность выбора данных для отчета, корректность формата и содержания отчета, и успешность его генерации.
10. Проведите тестирование на соответствие требованиям качества, проверяющее полноту и корректность документ

ПК 1.5

1. Изучите код и улучшите его структуру, сокращая повторяющийся код и упрощая сложные условия.

2. Оптимизируйте код для уменьшения времени выполнения, используя более эффективные алгоритмы или оптимизируя циклы и операции.
3. Примените шаблоны проектирования, такие как Singleton, Factory или Decorator, чтобы упростить и улучшить код.
4. Оптимизируйте использование памяти, уменьшая количество выделений памяти и удаляя утечки памяти.
5. Используйте инструменты профилирования кода, чтобы определить узкие места в коде и оптимизировать их.
6. Примените соглашения об именовании переменных и методов, чтобы улучшить читаемость и понимание кода.
7. Используйте инкапсуляцию и наследование, чтобы упростить код и избежать дублирования.
8. Проведите рефакторинг кода, разбивая его на более мелкие и понятные модули.
9. Используйте библиотеки и стандартные классы, чтобы избежать написания собственных решений и упростить код.
10. Оптимизируйте код для повышения производительности, например, улучшая работу с кэшем, сокращая количество запросов к базе данных и т. д.

ПК 1.6

1. Разработайте модуль, который использует API, чтобы получать данные о погоде и выводить их на экран устройства.
2. Создайте модуль для мобильного приложения, который позволяет пользователям искать и просматривать информацию о фильмах.
3. Разработайте модуль для мобильного приложения, который использует геолокацию, чтобы предлагать пользователю ближайшие к нему места для посещения.
4. Создайте модуль для мобильного приложения, который позволяет пользователям взаимодействовать с базой данных, сохраняя и извлекая информацию о задачах.
5. Разработайте модуль для мобильного приложения, который использует сенсоры устройства, чтобы предоставлять информацию о физической активности пользователя.
6. Создайте модуль для мобильного приложения, который позволяет пользователям создавать и редактировать свои списки покупок.
7. Разработайте модуль для мобильного приложения, который предоставляет возможность пользователю отправлять и принимать сообщения через SMS или электронную почту.
8. Создайте модуль для мобильного приложения, который использует функцию распознавания речи, чтобы предоставлять пользователю возможность управлять устройством голосом.
9. Разработайте модуль для мобильного приложения, который позволяет пользователям создавать и просматривать свои фотографии и видео.
10. Создайте модуль для мобильного приложения, который использует функцию оплаты, чтобы позволить пользователям оплачивать товары и услуги через приложение.

«Пермский национальный исследовательский политехнический университет»

«Сопровождение и обслуживание программного обеспечения компьютерных систем»

ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

ПМ.01. РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ

По специальности 09.02.07 Информационные системы и программирование

гр.

[illegible]

Председатель экзаменационной комиссии: _____ / _____ / « _____ » _____ **20** ____ г.

Члены экзаменационной комиссии: _____ / _____ / « _____ » _____ **20** ____ г.

_____ / _____ / « _____ » _____ **20** ____ г.

_____ / _____ / « _____ » _____ **20** ____ г.

2 Показатели и методы оценки экзамена по модулю ПМ.04 Сопровождение и обслуживание программного обеспечения компьютерных систем

Код и наименование профессиональных и общих компетенций, формируемых в рамках ПМ	Основные показатели оценки результата	Методы оценивания
ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием	Знание: – Основных этапов разработки программного обеспечения. – Основных принципов технологии структурного и объектно-ориентированного программирования. – Актуальной нормативно-правовой базы в области документирования алгоритмов. Умение: – Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. – Оформлять документацию на программные средства. – Оценка сложности алгоритма Практический опыт: – Разрабатывать алгоритм решения поставленной задачи и реализовывать его средствами автоматизированного проектирования	<i>Экзамен по модулю</i> <i>Аттестационный листы по практикам</i> <i>Экспертная оценка по результатам наблюдения за деятельностью обучающегося в процессе освоения ПМ</i>
ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием	Знание: – Основные этапы разработки программного обеспечения. – Основные принципы технологии структурного и объектно-ориентированного программирования. – Знание API современных мобильных операционных систем. Умение: – Создавать программу по разработанному алгоритму как отдельный модуль. – Оформлять документацию на	

	программные средства. – Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней, в том числе для мобильных платформ Практический опыт: – Разрабатывать код программного продукта на основе готовой спецификации на уровне модуля. – Разрабатывать мобильные приложения	
ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.	Практический опыт: – Использовать инструментальные средства на этапе отладки программного продукта. – Проводить тестирование программного модуля по определенному сценарию. Умения: – Выполнять отладку и тестирование программы на уровне модуля. – Оформлять документацию на программные средства. – Применять инструментальные средства отладки программного обеспечения. Знания: – Основные принципы отладки и тестирования программных продуктов – Инструментарий отладки программных продуктов.	
ПК 1.4. Выполнять тестирование программных модулей.	Практический опыт: – Проводить тестирование программного модуля по определенному сценарию. – Использовать инструментальные средства на этапе тестирования программного продукта. Умения:	

	– Выполнять отладку и тестирование программы на уровне модуля. – Оформлять документацию на программные средства. Знания: – Основные виды и принципы тестирования программных продуктов.	
ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.	Практический опыт: – Анализировать алгоритмы, в том числе с применением инструментальных средств. – Осуществлять рефакторинг и оптимизацию программного кода. Умения: – Выполнять оптимизацию и рефакторинг программного кода. – Работать с системой контроля версий. Знания: – Способы оптимизации и приемы рефакторинга. – Инструментальные средства анализа алгоритма. – Методы организации рефакторинга и оптимизации кода. – Принципы работы с системой контроля версий.	
ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.	Практический опыт: – Разрабатывать мобильные приложения. Умения: – Осуществлять разработку кода программного модуля на современных языках программирования. – Оформлять документацию на программные средства. Знания:	

	– Основные этапы разработки программного обеспечения. – Основные принципы технологии структурного и объектно-ориентированного программирования.	
--	--	--

3 Критерии оценивания экзамена по модулю

Критерии	Оценка
выполнено $\geq 80\%$ заданий И при выполнении заданий допущено ≤ 3 ошибок И задания выполнены в установленные сроки	Зачтено
выполнено $\geq 100\%$ заданий И при выполнении заданий допущено ≤ 5 ошибок И задания выполнены в установленные сроки	Зачтено
выполнено $\leq 80\%$ заданий ИЛИ при выполнении заданий допущено ≥ 5 ошибок	Не зачтено

4 Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Печатные издания

Основные источники

1. Федорова Г.Н. Разработка программных модулей программного обеспечения для компьютерных систем: учебник. Среднее профессиональное образование, профессиональная подготовка / Г.Н Федорова. – М.: Академия, 2018. – 336 с.

Дополнительные источники

1. Подбельский В. Язык С#. Базовый курс. Издание второе, переработанное и дополненное. Издательство: Финансы и статистика, 2013. – 408 с. -

Периодические издания

1. Вестник ПНИПУ. Электротехника, информационные технологии, системы управления [Текст]: научный рецензируемый журнал. Архив номеров 2010-2020 гг. – Режим доступа: <http://vestnik.pstu.ru/elinf/about/inf/> , свободный

2. Мир ПК: журнал для пользователей персональных компьютеров/Учредитель InternationalDataGroup. – Архив номеров в фонде ОНБ ЛФ ПНИПУ 2011–2018 гг.

3. Chip: журнал информационных технологий/Учредитель и издатель ЗАО «Издательский Дом Бурда». – Архив номеров в фонде ОНБ ЛФ ПНИПУ 2011–2018 гг.

4. Программные продукты и системы Издательство Научно-исследовательский институт «Центрпрограммсистем» . Архив номеров с 1988-2023 гг. Режим доступа: <https://e.lanbook.com/journal/2276?category=1537,авторизованный>

5. Системный администратор: ежемесячный журнал; включен в перечень ведущих рецензируемых журналов ВАК Минобрнауки РФ/Издатель ООО «ИД Положевец и партнеры». – Архив номеров в фонде ОНБ ЛФ ПНИПУ 2013-2017 гг.

6. Научно-технический и научно-производственный журнал Информационные технологии Издательство «Новые технологии» Эл. архив номеров с 2002-по 2023 Режим доступа: <http://novtex.ru/IT/> , свободный

Электронные издания (электронные ресурсы)

Основные источники

1. Белугина, С. В. Разработка программных модулей программного обеспечения для компьютерных систем. Прикладное программирование : учебное пособие / С. В. Белугина. —

Санкт-Петербург : Лань, 2020. — 312 с. — Режим доступа: <https://e.lanbook.com/book/133920> , авторизованный

Дополнительные источники

1. Васильев, Н. П. Введение в гибридные технологии разработки мобильных приложений : учебное пособие / Н. П. Васильев, А. М. Заяц. — Санкт-Петербург : Лань, 2020. — 160 с. — Режим доступа: <https://e.lanbook.com/book/147134>, авторизованный

2. Зубкова, Т. М. Технология разработки программного обеспечения : учебное пособие / Т. М. Зубкова. — Санкт-Петербург : Лань, 2019. — 324 с. — Режим доступа: <https://e.lanbook.com/book/122176> , авторизованный

3. Жулабова, Ф. Т. Системное программирование. Лабораторные работы : учебное пособие / Ф. Т. Жулабова. — Санкт-Петербург : Лань, 2020. — 208 с. — Режим доступа: <https://e.lanbook.com/book/140772> , авторизованный

4. Кузнецов, А. С. Системное программирование : учебное пособие / А. С. Кузнецов, И. А. Якимов, П. В. Пересунько. — Красноярск : СФУ, 2018. — 170 с. — Режим доступа: <https://e.lanbook.com/book/157574> , авторизованный

5. Лауферман, О. В. Разработка программного продукта: профессиональные стандарты, жизненный цикл, командная работа : учебное пособие / О. В. Лауферман, Н. И. Лыгина. — Новосибирск : Новосибирский государственный технический университет, 2019. — 75 с. — Режим доступа: <https://www.iprbookshop.ru/99215.html> , авторизованный

6. Маляров, А. Н. Объектно-ориентированное программирование : учебник для технических вузов / А. Н. Маляров. — Самара : Самарский государственный технический университет, ЭБС АСВ, 2017. — 332 с. — Режим доступа: <https://www.iprbookshop.ru/91772.html> , авторизованный

7. Забержинский, Б. Э. Программирование. Введение в разработку на С# : учебное пособие / Б. Э. Забержинский, А. Г. Золин. — Самара : Самарский государственный технический университет, ЭБС АСВ, 2017. — 120 с. — Режим доступа: <https://www.iprbookshop.ru/90876.html> , авторизованный

Интернет ресурсы

1. Учебники по программированию <http://programm.ws/index.php>
2. База знаний – Режим доступа: <https://habr.com/ru/all/>
3. Информационно-справочная система "КонсультантПлюс"

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ п.п.	Содержание изменения	Дата, номер протокола заседания ПЦК Подпись председателя ПЦК