

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Дарбинян Арсен Арамаисович  
Должность: Директор  
Дата подписания: 24.05.2023 16:58:36  
Уникальный программный ключ:  
e2db3198ab8522311de5257e53fe30e8fd1816c1



Частное профессиональное образовательное учреждение  
«Московский областной современный колледж»

**РАССМОТРЕНО И ОДОБРЕНО**  
на заседании Педагогического совета МОСК  
протокол № 2 от 22.05.2023 г.



**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
профессионального модуля

**ПМ.01 РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ  
ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ**

по специальности среднего профессионального образования  
**09.02.07 Информационные системы и программирование**

Подольск, 2023

Фонд оценочных средств по модулю **ПМ.01 Разработка модулей программного обеспечения для компьютерных систем** разработан на основе Федерального государственного образовательного стандарта среднего профессионального образования 09.02.07 Информационные системы и программирование, рабочей программы профессионального модуля, Положения о текущем контроле и промежуточной аттестации МОСК.

Организация-разработчик:

Частное профессиональное образовательное учреждение «Московский областной современный колледж» (МОСК)

## СОДЕРЖАНИЕ:

Общие положения .....	4
1. Формы контроля и оценивания элементов профессионального модуля.....	6
2. Результаты освоения модуля, подлежащие проверке на экзамене.....	7
3. Оценка освоения теоретического курса профессионального модуля «ПМ.01 Разработка модулей программного обеспечения для компьютерных систем» ....	15
3.1. Типовые задания для оценки освоения междисциплинарного курса «МДК.01.01 Разработка программных модулей» .....	16
3.2. Типовые задания для оценки освоения междисциплинарного курса «МДК.01.02 Поддержка и тестирование программных модулей».....	26
3.3. Типовые задания для оценки освоения междисциплинарного курса «МДК.01.03 Разработка мобильных приложений».....	32
3.4. Типовые задания для оценки освоения междисциплинарного курса «МДК.01.04 Системное программирование».....	32
4. Требования к дифференцированному зачету по учебной и производственной практике.....	48
4.1 Типовые задания для оценки умений и практического опыта по учебной практике.....	49
4.2 Типовые задания для оценки освоения профессиональных и общих компетенций по производственной практике .....	50
5. Структура контрольно-оценочных материалов для экзамена по модулю .....	50
6. Учебно-методическое и информационное обеспечение модуля.....	55
Приложение 1 .....	<b>Error! Bookmark not defined.</b>
Приложение 2.....	<b>Error! Bookmark not defined.</b>
Приложение 3.....	61
Приложение 4.....	62

## Общие положения

Результатом освоения профессионального модуля является готовность студента к выполнению вида профессиональной деятельности «ПМ.01 Разработка модулей программного обеспечения для компьютерных систем».

Формами аттестации по профессиональному модулю являются: дифференцированный зачет по междисциплинарному курсу «МДК.01.01 Разработка программных модулей», другая форма промежуточной аттестации и контрольная работа по междисциплинарному курсу «МДК.01.02 Поддержка и тестирование программных модулей», дифференцированный зачет по междисциплинарному курсу «МДК.01.03 Разработка мобильных приложений», экзамен по междисциплинарному курсу «МДК.01.04 Системное программирование», дифференцированный зачет по учебной практике, дифференцированный зачет по производственной практике и экзамен по профессиональному модулю.

Формой промежуточного контроля освоения междисциплинарных курсов является оценка выполнения контрольных заданий по внеаудиторной самостоятельной работе студентов.

Экзамены и дифференцированные зачеты по междисциплинарным курсам «МДК.01.01 Разработка программных модулей», «МДК.01.02 Поддержка и тестирование программных модулей», «МДК.01.03 Разработка мобильных приложений», «МДК.01.04 Системное программирование», проводятся в форме устного опроса или тестирования и решения практических задач.

Результаты дифференцированного зачета/ экзамена по междисциплинарному курсу отражаются в Сводной ведомости результатов освоения междисциплинарного курса (Приложения 1, 2).

Формой аттестации студентов по результатам учебной практики является дифференцированный зачет, предусматривающий защиту отчета по практике и решение практических задач. Результаты дифференцированного зачета регистрируются в Ведомости результатов дифференцированного зачета по практике (Приложение 3).

Формой аттестации студентов по результатам производственной практики также является дифференцированный зачет, предусматривающий защиту отчета по практике, решение практических задач. Условие допуска к зачету - успешное прохождение производственной практики. Виды и качество выполненных производственных работ отражается работодателем в аттестационном листе. Работодатель оформляет характеристику профессиональной деятельности студента в период производственной практики, которая учитывается при подведении итогов дифференцированного зачета. Результаты дифференцированного зачета по производственной практике регистрируются в ведомости (Приложение 3).

Экзамен по модулю предусматривает решение комплексной профессиональной задачи, задания которой ориентированы на проверку освоения предусмотренных профессиональных компетенций и вида

профессиональной деятельности в целом. Общие компетенции формируются в процессе освоения образовательной программы в целом, поэтому по результатам освоения профессионального модуля оценивается положительная динамика их формирования.

Результаты освоения компетенций регистрируются в оценочной ведомости (см. образец оформления - Приложение 4).

Итогом экзамена является однозначное решение: «вид профессиональной деятельности освоен/не освоен».

## 1. Формы контроля и оценивания элементов профессионального модуля

Модуль и его элементы	Форма контроля и оценивания	
	Промежуточная аттестация	Промежуточный контроль
МДК.01.01 Разработка программных модулей	Другая форма контроля, защита курсовой работы, экзамен	Оценка выполненных контрольных заданий внеаудиторной самостоятельной работы
МДК.01.02 Поддержка и тестирование программных модулей	Другая форма контроля	Оценка выполненных контрольных заданий внеаудиторной самостоятельной работы
МДК.01.03 Разработка мобильных приложений	Другая форма контроля, Дифференцированный зачет	Оценка выполненных контрольных заданий внеаудиторной самостоятельной работы
МДК.01.04 Системное программирование	Дифференцированный зачет	Оценка выполненных контрольных заданий внеаудиторной самостоятельной работы
УП.01.01. Учебная практика	Дифференцированный зачет	Оценка практических умений, аттестационный лист студента по УП
ПП.01.01. Производственная практика (по профилю специальности)	Дифференцированный зачет	Оценка освоения профессиональных и динамики освоения общих компетенций, Аттестационный лист студента по ПП, характеристика профессиональной деятельности студента в период производственной практики
Профессиональный модуль «ПМ.01 Разработка модулей программного обеспечения для компьютерных систем»	Экзамен по модулю	Оценка результатов решения профессиональной задачи, экспертное заключение работодателя.

## 2. Результаты освоения модуля, подлежащие проверке на экзамене

### 2.1 Профессиональные и общие компетенции

В результате аттестации по профессиональному модулю осуществляется комплексная проверка следующих профессиональных и общих компетенций:

Код и наименование профессиональных и общих компетенций, формируемых в рамках модуля <sup>1</sup>	Критерии оценки	Методы оценки
<b>Раздел модуля 1. Разработка программных модулей</b>		
ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием	<p>Оценка <b>«отлично»</b> - техническое задание проанализировано, алгоритм разработан, соответствует техническому заданию и оформлен в соответствии со стандартами, пояснены его основные структуры. указаны использованные стандарты в области документирования; выполнена оценка сложности алгоритма</p> <p>Оценка <b>«хорошо»</b> - алгоритм разработан, оформлен в соответствии со стандартами и соответствует заданию, пояснены его основные структуры. выполнена оценка сложности алгоритма</p> <p>Оценка <b>«удовлетворительно»</b> - алгоритм разработан и соответствует заданию.</p>	<p>Экзамен/зачет в форме собеседования: практическое задание по построению алгоритма в соответствии с техническим заданием</p> <p>Защита отчетов по практическим и лабораторным работам</p>
ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием	<p>Оценка <b>«отлично»</b> - программный модуль разработан по имеющемуся алгоритму в среде разработки (на указанном языке программирования) методами объектно-ориентированного/структурного программирования и полностью соответствует техническому заданию, соблюдены и пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка <b>«хорошо»</b> - программный модуль разработан по имеющемуся алгоритму в среде разработки (указанном языке программирования) методами объектно-ориентированного/структурного программирования и практически соответствует техническому заданию с незначительными отклонениями, пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка <b>«удовлетворительно»</b> -</p>	<p>Экзамен/зачет в форме собеседования: практическое задание по построению алгоритма в соответствии с техническим заданием</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p> <p>Тестирование</p>

<sup>1</sup> В ходе оценивания могут быть учтены личностные результаты

	программный модуль разработан по имеющемуся алгоритму в среде разработки (на указанном языке программирования) методами объектно-ориентированного/структурного программирования и соответствует техническому заданию; документация на модуль оформлена без существенных отклонений от стандартов.	
<b>Раздел модуля 2. Поддержка и тестирование программных модулей</b>		
ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств	<p>Оценка <b>«отлично»</b> - выполнена отладка модуля (с использованием инструментария среды проектирования); с пояснением особенностей отладочных классов; сохранены и представлены результаты отладки.</p> <p>Оценка <b>«хорошо»</b> - выполнена отладка модуля (с использованием инструментария среды проектирования); сохранены и представлены результаты отладки.</p> <p>Оценка <b>«удовлетворительно»</b> - выполнена отладка модуля, пояснены ее результаты.</p>	<p>Экзамен/зачет в форме собеседования: практическое задание по построению алгоритма в соответствии с техническим заданием</p> <p>Защита отчетов по практическим и лабораторным работам Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики Тестирование</p>
ПК 1.4 Выполнять тестирование программных модулей	<p>Оценка <b>«отлично»</b> - выполнено тестирование модуля, в том числе с помощью инструментальных средств, и оформлены результаты тестирования в соответствии со стандартами.</p> <p>Оценка <b>«хорошо»</b> - выполнено тестирование модуля, в том числе с помощью инструментальных средств, и оформлены результаты тестирования.</p> <p>Оценка <b>«удовлетворительно»</b> - выполнено тестирование модуля и оформлены результаты тестирования.</p>	<p>Экзамен/зачет в форме собеседования: практическое задание по выполнению заданных видов тестирования программного модуля. тестирование</p> <p>Защита отчетов по практическим и лабораторным работам Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p>
ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода	<p>Оценка <b>«отлично»</b> - определены качественные характеристики программного кода с помощью инструментальных средств; выявлены фрагменты некачественного кода; выполнен рефакторинг на уровнях переменных, функций, классов, алгоритмических структур; проведена оптимизация и подтверждено повышение качества программного кода.</p> <p>Оценка <b>«хорошо»</b> - определены</p>	<p>Экзамен/зачет в форме собеседования: практическое задание по оценке качества кода предложенного программного модуля, поиску некачественного программного кода, его анализу, оптимизации методами рефакторинга.</p>



	<p>качественные характеристики программного кода с помощью инструментальных средств; выявлены фрагменты некачественного кода; выполнен рефакторинг на нескольких уровнях; проведена оптимизация и выполнена оценка качества полученного программного кода.</p> <p>Оценка «удовлетворительно» - определены качественные характеристики программного кода частично с помощью инструментальных средств; выявлено несколько фрагментов некачественного кода; выполнен рефакторинг на нескольких уровнях; проведена оптимизация и выполнена оценка качества полученного программного кода.</p>	<p>Защита отчетов по практическим и лабораторным работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p> <p>Тестирование</p>
<b>Раздел модуля 3. Разработка мобильных приложений</b>		
<p>ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием</p>	<p>Оценка «отлично» - программный модуль разработан по имеющемуся алгоритму в среде разработки (на указанном языке программирования) методами объектно-ориентированного/структурного программирования и полностью соответствует техническому заданию, соблюдены и пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка «хорошо» - программный модуль разработан по имеющемуся алгоритму в среде разработки (на указанном языке программирования) методами объектно-ориентированного/структурного программирования и практически соответствует техническому заданию с незначительными отклонениями, пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка «удовлетворительно» - программный модуль разработан по имеющемуся алгоритму в среде разработки (на указанном языке программирования) методами объектно-ориентированного/структурного программирования и соответствует техническому заданию; документация на модуль оформлена без существенных отклонений от стандартов</p>	<p>Экзамен/зачет в форме собеседования:</p> <p>практическое задание по разработке программного модуля в соответствии с техническим заданием</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p> <p>Тестирование</p>
<p>ПК 1.6 Разрабатывать модули программного</p>	<p>Оценка «отлично» - разработан модуль для заданного мобильного устройства с</p>	<p>Экзамен/зачет в форме собеседования:</p>

<p>обеспечения для мобильных платформ.</p>	<p>соблюдением основных этапов разработки на одном из современных языков программирования; при проверке работоспособности модуля на устройстве или эмуляторе установлено его соответствие спецификации.</p> <p>Оценка «хорошо» - разработан модуль для заданного мобильного устройства с учетом основных этапов разработки на одном из современных языков программирования; при проверке работоспособности модуля на устройстве или эмуляторе установлено соответствие выполняемых функций спецификации с незначительными отклонениями.</p> <p>Оценка «удовлетворительно» - разработан модуль для заданного мобильного устройства на одном из современных языков программирования; при проверке работоспособности модуля на устройстве или эмуляторе установлено соответствие основных выполняемых функций спецификации.</p>	<p>практическое задание по созданию модуля для заданного мобильного устройства на основе спецификации</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p> <p>Тестирование</p>
<p><b>Раздел модуля 4. Системное программирование</b></p>		
<p>ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием</p>	<p>Оценка «отлично» - программный модуль разработан по имеющемуся алгоритму в среде разработки (на указанном языке программирования) методами объектно-ориентированного/структурного программирования и полностью соответствует техническому заданию, соблюдены и пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка «хорошо» - программный модуль разработан по имеющемуся алгоритму в среде разработки (на указанном языке программирования) методами объектно-ориентированного/структурного программирования и практически соответствует техническому заданию с незначительными отклонениями, пояснены основные этапы разработки; документация на модуль оформлена и соответствует стандартам.</p> <p>Оценка «удовлетворительно» - программный модуль разработан по имеющемуся алгоритму в среде разработки (на указанном языке программирования) методами объектно-ориентированного/структурного программирования и соответствует</p>	<p>Экзамен/зачет в форме собеседования:</p> <p>практическое задание по разработке программного модуля в соответствии с техническим заданием</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p> <p>Тестирование</p>

	техническому заданию; документация на модуль оформлена без существенных отклонений от стандартов.	
ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств	<p>Оценка «отлично» - выполнена отладка модуля (с использованием инструментария среды проектирования); с пояснением особенностей отладочных классов; сохранены и представлены результаты отладки.</p> <p>Оценка «хорошо» - выполнена отладка модуля (с использованием инструментария среды проектирования); сохранены и представлены результаты отладки.</p> <p>Оценка «удовлетворительно» - выполнена отладка модуля, пояснены ее результаты.</p>	<p>Экзамен/зачет в форме собеседования:</p> <p>практическое задание по выполнению отладки предложенного программного модуля</p> <p>Защита отчетов по практическим и лабораторным работам</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики</p> <p>Тестирование</p>
ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.	<p>– обоснованность постановки цели, выбора и применения методов и способов решения профессиональных задач;</p> <p>- адекватная оценка и самооценка эффективности и качества выполнения профессиональных задач</p>	<p>Экспертное наблюдение за выполнением работ, тестирование</p>
ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.	- использование различных источников, включая электронные ресурсы, медиаресурсы, Интернет-ресурсы, периодические издания по специальности для решения профессиональных задач	
ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.	<p>- демонстрация ответственности за принятые решения</p> <p>- обоснованность самоанализа и коррекция результатов собственной работы;</p>	
ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.	<p>- взаимодействовать с обучающимися, преподавателями и мастерами в ходе обучения, с руководителями учебной и производственной практик;</p> <p>- обоснованность анализа работы членов команды (подчиненных)</p>	
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и	Демонстрировать грамотность устной и письменной речи, - ясность формулирования и изложения мыслей	

культурного контекста.		
ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения.	- соблюдение норм поведения во время учебных занятий и прохождения учебной и производственной практик, соблюдение стандартов антикоррупционного поведения	
ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.	- эффективное выполнение правил ТБ во время учебных занятий, при прохождении учебной и производственной практик; - демонстрация знаний и использование ресурсосберегающих технологий в профессиональной деятельности	
ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.	- эффективность использовать средств физической культуры для сохранения и укрепления здоровья при выполнении профессиональной деятельности.	
ОК 09. Использовать информационные технологии в профессиональной деятельности.	- эффективность использования информационно-коммуникационных технологий в профессиональной деятельности согласно формируемым умениям и получаемому практическому опыту;	
ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языках.	- эффективность использования в профессиональной деятельности необходимой технической документации, в том числе на английском языке.	

## 2.2 Курсовая работа

### 2.2.1. Требования к курсовой работе:

1. Содержание курсовой работы должно раскрывать тему и соответствовать плану.
2. В курсовой работе необходимо отразить собственное понимание и

осмысление студентом проблемы на основе изучения литературы, оценку тех или других аспектов теорий и концепций со ссылкой на их авторов.

3. Курсовая работа должна носить творческий характер, позволяющий раскрыть потенциал студента, его эрудицию, кругозор, умение находить инновационные, нестандартные пути решения проблем.

4. Все разделы курсовой работы выстраиваются в строгой логической последовательности и взаимосвязанности. Материал внутри раздела также должен быть изложен логически последовательно. При его подаче необходимо использовать научный стиль русского литературного языка.

5. Курсовая работа выполняется с соблюдением орфографических, пунктуационных, стилистических норм русского языка.

### **2.2.2. Тематика курсовых работ:**

1. Разработка базы данных и приложения для типографии
2. Разработка базы данных и приложения для проката автомобиля
3. Разработка базы данных и приложения для проката инструментов
4. Разработка приложения «Вкладыш к диплому»
5. Разработка базы данных и приложения расписания мероприятий для детского сада
6. Разработка базы данных и клиентского приложения для отдела кадров
7. Разработка базы данных и приложения для продажи билетов в кассе автовокзала
8. Разработка клиентской части сайта цветочного магазина
9. Разработка базы данных и приложения «Личное дело»
10. Разработка модуля приложения «Умный дом»
11. Разработка серверной части сайта строительной компании
12. Разработка клиентской части сайта агентства недвижимости
13. Разработка базы данных и приложения для заказа билетов в кинотеатры
14. Разработка утилиты перевода системы документооборота на новый отчетный период
15. Разработка базы данных и приложения для заказа товаров в магазине строительных товаров
16. Разработка клиент серверного приложения модуля «Учебные материалы»
17. Разработка клиентской части приложения для медицинского работника
18. Разработка базы данных и приложения «расписание»
19. Разработка базы данных учёта материалов и товаров
20. Разработка клиентской части сайта продажи автозапчастей
21. Разработка клиентской части сайта магазина спортивной одежды
22. Разработка клиентского части сайта для продуктового магазина
23. Разработка базы данных и административной части Интернет-магазина электроники
24. Разработка серверной части сайта жилищной. компании
25. Разработка базы данных и приложения учета товаров на складе
26. Разработка клиентского приложения «Классный руководитель»
27. Разработка базы данных электронного учебника дисциплины
28. Разработка базы данных и приложения «Книгофонд»

29. Разработка базы данных и приложения для учета мероприятий службой по организации праздников
30. Разработка базы данных и приложения учета клиентов в гостинице
31. Разработка клиентского приложения сайта зоомагазина
32. Разработка базы данных и приложения учета работы пассажирского автотранспортного предприятия
33. Разработать базу данных и приложения работы автобусного парка
34. Разработка клиентского приложения для продуктового магазина
35. Разработать базу данных и приложения учета оплаты коммунальных услуг физическими лицами
36. Разработка информационная справочная система по туристической фирме
37. Разработка базы данных и приложения учета информации в аптеке
38. Разработка автоматизированной информационной системы спортивного комплекса
39. Разработка базы данных и серверной части сайта автошколы
40. Разработка автоматизированной информационной справочная системы приемной комиссии учебного заведения
41. Разработка базы данных и приложения методической службы учебного заведения
42. ВЕБ разработка информационной справочной системы салона красоты
43. Разработать базу данных и приложения учета работы лыжной базы
44. Разработка приложения «Калькулятор расчета услуг» на сайте организации
45. Разработка электронного учебника по предмету
46. Разработка автоматизированной информационной справочной система «Центр занятости населения».
47. Разработка клиентской части сайта «Транспортное агентство»
48. Автоматизированная информационная система бухгалтерия предприятия
49. Разработка базы данных и клиентской части интернет-магазина бытовой техники
50. Разработка клиентского приложения «Сессия»
51. Разработка базы данных и приложения учета товаров продукции на складе
52. Разработка базы данных и приложения «Эмиграция населения»
53. Разработка базы данных и приложения учета работы органов записи актов гражданского состояния (ЗАГС)
54. Разработка клиентской части сайта интернет-магазина детской одежды
55. Разработка базы данных и приложения по формированию временных пропусков в организации
56. Разработка модулей «Интерфейс инструментальной среды»
57. Разработка базы данных и приложения для предприятия в системе с конфигурацией 1С
58. Разработка БД и приложения интернет-магазина сотовых телефонов
59. Разработка базы данных и приложения учета заявок на авиабилеты
60. Разработка приложение учета заявок на покупку квартир и поиска вариантов.
61. Разработать приложение Учет заявок на авиабилеты

62. Автоматизированная информационная справочная система по туристической фирме.
63. Автоматизированная информационная справочная система городской больницы.
64. Автоматизированная информационная справочная система ГИБДД.
65. Автоматизированная информационная справочная система ГАОУ СПО «... техникум» - приемная комиссия.
66. Автоматизированная информационная справочная система «Центр занятости населения».
67. Разработать систему «Прокат DVD-дисков».
68. Разработать базу данных и приложение для инвентаризационного учёта мебели в помещениях колледжа.
69. Разработка базы данных и административного приложения для налоговой инспекции
70. Разработка модулей «Интерфейс инструментальной среды»

### **3. Оценка освоения теоретического курса профессионального модуля «ПМ.01 Разработка модулей программного обеспечения для компьютерных систем»**

Предметом оценки освоения теоретического курса по профессиональному модулю «ПМ.01 Разработка модулей программного обеспечения для компьютерных систем» являются умения и знания междисциплинарных курсов «МДК.01.01 Разработка программных модулей», «МДК.01.02 Поддержка и тестирование программных модулей», «МДК.01.03 Разработка мобильных приложений», «МДК.01.04 Системное программирование».

Формой рубежного контроля освоения умений и знаний междисциплинарным курсам является оценка правильности выполнения контрольных заданий по внеаудиторной самостоятельной работе.

Формой промежуточной аттестации по оценке освоения программы междисциплинарных курсов «МДК.01.01 Разработка программных модулей», «МДК.01.02 Поддержка и тестирование программных модулей», «МДК.01.03 Разработка мобильных приложений», «МДК.01.04 Системное программирование» являются экзамен, дифференцированный зачет и контрольная работа. Экзамен, дифференцированный зачет, контрольная работа по междисциплинарным курсам предусматривает оценку освоения знаний в форме устного опроса, тестирования и оценку освоения умений в результате решения практических задач.

Оценка освоения умений и знаний осуществляется на основе разработанных критериев.

Критерии оценивания заданий

5 «отлично» - глубоко и прочно усвоен весь программный материал по

модулю; последовательно и точно построена речь; теория увязывается с практической и профессиональной деятельностью; отсутствуют затруднения с ответами на дополнительные или уточняющие вопросы;

4 «хорошо» - усвоен весь программный материал; в речи имеются незначительные неточности; правильно применены теоретические знания; на большинство дополнительных или уточняющих вопросов дан ответ;

3 «удовлетворительно» - усвоена основная часть программного материала; речь не содержит «деталей»; недостаточно-правильные формулировки; затруднения в выполнении практических заданий; на большинство дополнительных или уточняющих вопросов испытываются затруднения в ответе;

2 «неудовлетворительно» - не усвоена значительная часть программного материала; ответ содержит существенные ошибки; затруднения в выполнении практических заданий, в формулировании основных дефиниций по курсу.

### **3.1. Типовые задания для оценки освоения междисциплинарного курса «МДК.01.01 Разработка программных модулей»**

#### **3.1.1. Типовые задания в тестовой форме для оценки знаний**

1. Какие программы можно отнести к системному программному обеспечению:

- а) прикладные программы;
- б) операционные системы;**
- в) игровые программы.

2. Можно ли отнести операционную систему к программному обеспечению:

- а) да;**
- б) нет.

3. Самый большой этап в жизненном цикле программы:

- а) изучение предметной области;
- б) программирование;
- в) тестирование;
- г) эксплуатация;**
- д) корректировка ошибок.

4. Какой этап выполняется раньше:

- а) отладка;**
- б) тестирование.

5. Способы оценки качества:

- а) наличие документации;
- б) сравнение с аналогами;**
- в) оптимизация программы;
- г) структурирование алгоритма.

6. Существует ли связь между эффективностью и оптимизацией программы:

- а) да;**



б) нет.

7. Можно ли внутри цикла поместить еще один цикл:

а) да;

б) нет.

8. Можно ли ставить знак подчеркивания в начале имени:

а) да, без ограничений;

б) да, но не рекомендуется;

в) нет.

9. Как называется способ составления имен переменных, когда в начале имени сообщается тип переменной:

а) прямым указанием;

б) венгерской нотацией;

в) структурным программированием;

г) поляризацией.

10. Можно ли писать комментарии в отдельной строке:

а) да;

б) нет.

11. Наличие комментариев позволяет:

а) быстрее писать программы;

б) быстрее выполнять программы.

в) быстрее найти ошибки в программе.

12. Возможно ли комбинирование языков программирования в рамках одной задачи:

а) нет.

б) да;

13. Для решения инженерных задач характерно применение:

а) САПР (систем автоматизированного проектирования);

б) СУБД (систем управления базами данных);

в) ОС (операционных систем).

14. Причины синтаксических ошибок:

а) ошибки в исходных данных;

б) ошибки, допущенные на более ранних этапах;

в) плохое знание языка программирования;

г) неправильное применение процедуры тестирования.

15. Защитное программирование это:

а) встраивание в программу отладочных средств;

б) создание задач защищенных от копирования;

- в) разделение доступа в программе;
- г) использование паролей;

16. Отладка – это:

- а) определение списка параметров;
- б) правило вызова процедур (функций);
- в) процедура поиска ошибок, когда известно, что ошибка есть;**
- г) составление блок-схемы алгоритма.

17. Когда программист может проследить последовательность выполнения команд программы:

- а) при тестировании;
- б) при трассировке;**
- в) при компиляции;
- г) при выполнении программы;
- д) при компоновке.

18. На каком этапе создания программы могут появиться синтаксические ошибки:

- а) анализ требований;
- б) проектирование;
- в) программирование;**
- г) тестирование.

19. Позволяет ли автоматизация программирования всегда создавать эффективные программы:

- а) да.
- б) нет;**

20. Позволяет ли автоматизация программирования всегда создавать надежные программы:

- а) нет;
- б) да.**

21. Что легко поддается автоматизации:

- а) работа с файлами;
- б) сложные логические задачи;
- в) интерфейс;**
- г) алгоритмизация.

22. Что такое оптимизация программ:

- а) создание удобного интерфейса пользователя;
- б) улучшение работы существующей программы;**
- в) разработка модульной конструкции программы;
- г) применение методов объектно-ориентированного программирования.

23. Сущность оптимизации циклов:

- а) трассировка циклов;
- б) сокращение тела цикла;
- в) представление циклов в виде блок-схем;
- г) **сокращение количества повторений выполнения тела цикла**

24. В чем сущность модульного программирования:

- а) в разбиении программы на отдельные равные части;
- б) **в разбиении программы на отдельные функционально независимые части;**
- в) в разбиение программы на процедуры и функции;
- г) снижает количество ошибок.

25. Недостаток модульного программирования:

- а) увеличивает трудоемкость программирования;
- б) снижает быстродействие программы;
- в) не позволяет выполнять оптимизацию программы.
- г) **усложняет процедуру комплексного тестирования;**

26. При структурном программировании задача выполняется:

- а) поэтапным разбиением на более легкие задачи;
- б) без участия программиста;
- в) **объединением отдельных модулей программы.**

27. Достоинство структурного программирования:

- а) можно приступить к автономному тестированию на раннем этапе разработки;
- б) нет необходимости выполнять тестирование;
- в) **можно приступить к комплексному тестированию на раннем этапе разработки;**
- г) можно пренебречь отладкой.

28. Может ли дочерний элемент иметь двух родителей:

- а) да;
- б) **нет;**
- в) только для визуальных элементов;
- г) если их свойства совпадают.

29. Есть ли различие между объектом и экземпляром:

- а) **да;**
- б) нет;
- в) если у них общий предок.

30. Могут ли два экземпляра одного объекта реагировать на событие по-разному:

- а) **да;**

б) нет.

31. Какие этапы проектирования можно объединять:

- а) эскизный и рабочий;
- б) технический и эскизный.
- в) технический и рабочий;**

32. Процесс преобразования постановки задачи в план алгоритмического или вычислительного решения это:

- а) анализ требований;
- б) программирование;
- в) проектирование;**
- г) тестирование.

33. Модульное программирование применимо при:

- а) проектировании сверху вниз;
- б) проектирование снизу-вверх;**

34. Проектирование сверху вниз это:

- а) последовательное разбиение общих задач на более мелкие;**
- б) составление из отдельных модулей большой программы.

35. Проектирование снизу-вверх это:

- а) составление из отдельных модулей большой программы;**
- б) последовательное разбиение общих задач на более мелкие.

36. Зависит ли трудоемкость разработки от вида информации:

- а) да;**
- б) нет.

37. Кому принадлежит право собственности на ПО:

- а) продавцу;
- б) разработчику;**
- в) покупателю.

38. Кому принадлежит авторское право на ПО:

- а) покупателю.
- б) продавцу;
- в) разработчику;**

39. Если вы приобрели программы законным путем, имеете ли вы право продать ее:

- а) да;**
- б) нет.

40. Если вы приобрели программу законным путем, являетесь ли вы

собственником программы:

- а) нет;
- б) да.

### **3.1.2 Задания для выполнения внеаудиторной самостоятельной работы**

1. Создать форму для ввода массива и кнопкой «Перемешать». При нажатии на кнопку элементы массива перемешиваются
2. Палиндромом называют последовательность символов, которая читается как слева направо, так и справа налево. Создать web-форму для ввода строки и кнопкой, при нажатии на которую определяется, является ли введенная строка палиндромом
3. Создать web-форму с двумя полями для ввода чисел и выбором операции (+, -, \*, /). Вычислить соответствующее выражение и вывести результат в браузер
4. Найти наибольшее из трех чисел, введенных пользователем в форму
5. Написать программу нахождения синуса и косинуса угла, выбор действия осуществляется с помощью переключателя
6. Средствами PHP вывести время и дату на странице
7. Создать web-форму для ввода строки и кнопкой, при нажатии на которую строка переворачивается и выводится в браузер
8. Написать программу, которая для введенной строки подсчитывает сумму всех содержащихся в ней цифр
9. Создать класс Point, разработав следующие элементы класса: поля (x, y); конструкторы, позволяющие создать экземпляр класса с нулевыми координатами или с заданными координатами; методы, позволяющие вывести координаты точки на экран, рассчитать расстояние от начала координат до точки, переместить точку на плоскости на вектор (a, b); свойства для получения-установки координаты точки (доступное для чтения и записи), для умножения координаты точки на скаляр (доступное только для записи)
10. Создать класс Triangle, разработав следующие элементы класса: поля (a, b, c); конструктор, позволяющий создать экземпляр класса с заданными длинами сторон; методы, позволяющие вывести длины сторон треугольника на экран, рассчитать периметр треугольника, рассчитать площадь треугольника; свойства для получения-установки длин сторон треугольника (доступное для чтения и записи), для определения, существует ли треугольник с данными длинами сторон (доступное только для чтения)
11. Создать класс Rectangle, разработав следующие элементы класса: поля (a, b); конструктор, позволяющий создать экземпляр класса с заданными длинами сторон; методы, позволяющие вывести длины сторон прямоугольника на экран, рассчитать периметр прямоугольника, рассчитать площадь прямоугольника; свойства для получения-установки длин сторон прямоугольника (доступное для чтения и записи), для определения, является ли данный прямоугольник квадратом (доступное только для чтения)
12. Создать класс Round, разработав следующие элементы класса: поля (r); конструктор, позволяющий создать экземпляр класса с заданным радиусом; методы, позволяющие рассчитать длину окружности, рассчитать площадь круга; свойства для получения-установки радиуса круга (доступное для чтения и

записи)

13. Создать абстрактный класс Figure с методами вычисления площади и периметра, а также методом, выводящим информацию о фигуре на экран. Создать производные классы: Triangle (треугольник), Circle (круг) со своими методами вычисления площади и периметра. Создать массив n фигур и вывести полную информацию о фигурах на экран

14. Посчитать сумму элементов побочной диагонали квадратной матрицы

15. Найти среднее арифметическое положительных элементов одномерного массива. Количество элементов массива вводится пользователем, массив заполняется случайными числами

16. Посчитать в строке количество цифр. Строка вводится пользователем с клавиатуры

17. Создать форму Windows. Программа должна предоставлять возможность менять цвет фона формы

18. Создать Windows-приложение, демонстрирующее работу с модальными и немодальными окнами

19. Написать программу на языке C++, которая считает сумму элементов одномерного массива, не используя операцию индексирования [].

20. Написать программу на языке C++, демонстрирующую способы передачи параметров в функции

21. Написать рекурсивную функцию для нахождения факториала числа

22. Написать программу на языке C++ для подсчета суммы элементов одномерного массива. Размерность массива вводится пользователем с клавиатуры

23. Написать программу, демонстрирующую возможность создания односвязного списка и добавления элемента в список

24. Написать программу, демонстрирующую возможность создания стека и добавления элемента в любое место стека

25. Создать класс «Прямоугольник», который имеет два конструктора. Первый конструктор принимает в качестве параметров длину и ширину. Второй конструктор создает квадрат с заданной стороной

26. Написать программу, демонстрирующую работу с абстрактными классами

27. Создать функцию на языке C++, подсчитывающую сумму элементов массива. Массив передается как параметр в функцию

28. Написать программу для нахождения максимального элемента матрицы.

29. Написать программу для сортировки массива методом пузырька.

30. Написать программу для нахождения корней квадратного уравнения  $ax^2+bx+c=0$ . Коэффициенты a, b, c вводятся пользователем

Темы рефератов (докладов, презентаций)

1. История развития CASE-средств разработки программного обеспечения

2. Диаграммы развертывания

3. Методология разработки ПО RUP (Rational Unified Process)

4. Методология разработки ПО Crystal Clear

5. Среда dotNet (.NetFramework)

6. Измерение и оценка характеристик качества ПО

7. Роль стандартизации и сертификации в управлении качеством ПС

8. Виды тестирования ПО
9. Тестирование объектно ориентированных систем
10. Тестирование Веб-приложений
11. Принципы обеспечения безопасности ПО на различных стадиях его жизненного цикла: при планировании работ и проектном анализе ПО, в процессе его разработки, при приемо-сдаточных испытаниях, при эксплуатации ПО
12. Принципы криптографической защиты информации
13. Технологии аутентификации
14. Правовые методы защиты программных продуктов и баз данных
15. Сопровождение программного средства

### **3.1.3. Типовые задания для дифференцированного зачета по междисциплинарному курсу «МДК.01.01 Разработка программных модулей»**

1. Понятие жизненного цикла ПО.
2. Этапы жизненного цикла ПО.
3. Стадия предпроектных исследований и технического предложения (анализ) — определение методов решения задачи.
4. Стадия эскизного проектирования — разработка структуры программного продукта, выбор структур для хранения данных, построение и оценка алгоритмов подпрограмм и определение особенностей взаимодействия программы с вычислительной средой (другими программами, программной системой и техническими средствами).
5. Стадия технического проектирования — составление программы на выбранном языке программирования, ее тестирование и отладка.
6. Стадия рабочего проектирования — оформление документации.
7. Стадия испытаний — всестороннее тестирование программы.
8. Стадия внедрения в эксплуатацию.
9. Методы программирования: структурный, модульный, объектно-ориентированный.
10. Достоинства и недостатки методов программирования.
11. Общие принципы разработки программного обеспечения.
12. Стадии и жизненный цикл программного обеспечения.
13. Модели жизненного цикла ПО.
14. Типы приложений.
15. Консольные приложения.
16. Оконные Windows приложения.
17. Web-приложения.
18. Библиотеки.
19. Web-сервисы.
20. Технология структурного программирования.
21. Инструментальные средства оформления и документирования алгоритмов программ.
22. Оценка сложности алгоритма: классификация, классы алгоритмов,

неразрешимые задачи.

23. Модульное программирование.

24. Объектно-ориентированное программирование.

25. Базовые принципы ООП: инкапсуляция, наследование, полиморфизм.

26. Классы: основные понятия.

27. Операции класса.

28. Иерархия классов.

29. Перегрузка методов.

30. Синтаксис интерфейсов.

31. Интерфейсы и наследование.

32. Структуры.

33. Диалекты.

34. Регулярные выражения.

35. Коллекции.

36. Параметризованные классы.

37. Указатели.

38. Операции со списками

1. Назначение и виды паттернов.

2. Основные шаблоны.

3. Порождающие шаблоны.

4. Структурные шаблоны.

5. Поведенческие шаблоны.

6. Событийно – управляемое программирование.

7. Элементы управления.

8. Диалоговые окна.

9. Обработчики событий.

10. Введение в графику.

11. Методы оптимизации программного кода.

12. Цели и методы рефакторинга.

### **3.1.4. Типовые задания для экзамена по междисциплинарному курсу «МДК.01.01 Разработка программных модулей»**

1. Особенности создания программного продукта. Проблематика проектирования.

2. Понятие технологии разработки программы. Модели жизненного цикла.

3. Унифицированный язык моделирования (UML). Определение прецедентов (вариантов использования).

4. Определение функциональных и эксплуатационных требований к программным продуктам при структурном и объектном подходе.

5. Словарь терминов. Функциональные диаграммы. Диаграммы потоков данных (DFD).

6. Проектирование программного обеспечения при структурном подходе.

7. Структурная схема разрабатываемого программного обеспечения (логическая и физическая модели)

8. Метод пошаговой детализации при составлении алгоритмов.



9. Пользовательские типы данных: структуры (записи), объединения, перечисления, множества.
10. Синтаксис объявления пользовательских типов данных. Доступ к элементу пользовательского типа.
11. Основы файловой системы. Порядок работы с текстовыми и двоичными файлами.
12. Функции для работы с файлами разного типа. Правила бесформатного и форматного ввода/вывода.
13. Разделение программы на подпрограммы. Параметры подпрограммы. Перегрузка подпрограмм.
14. Группировка подпрограмм и связанных с ними данных в отдельные файлы - модули. Основные характеристики программного модуля.
15. Модульная структура программных продуктов.
16. Проектирование программного обеспечения при объектном подходе.
17. Структура проекта в объектно-ориентированной среде программирования. Событийно-управляемая модель программирования.
18. Компоненты объектно-ориентированной среды. Принципы разработки графического интерфейса.
19. Особенности ввода/вывода в объектно-ориентированной среде программирования.
20. Классы. Конструкторы и деструкторы. Наследование. Создание наследованного класса.
21. Статический и динамический полиморфизм (виртуализация методов). Дружественные функции и классы.
22. Механизм обработки и генерирования исключительных ситуаций. Способы создания DLL-библиотеки.
23. Серия стандартов ISO 9000. Процесс сертификации программ на базе информации об их использовании.
24. Виды программных документов. Пояснительная записка. Руководство пользователя. Руководство системного программиста.
25. Тестирование методами «белого ящика» и «черного ящика».
26. Порядок разработки тестов. Автоматизация тестирования.
27. Модульное, интеграционное, системное тестирование.
28. Отладка программ. Автономная и комплексная отладка программ.
29. Инструментальные средства отладки программ.
30. Основные принципы построения приложений, основанных на СУБД.
31. Компоненты объектно-ориентированной среды, необходимые для организации базы данных и работы с ней.
32. Принцип разработки программы, управляющей локальной базой данных. Способы навигации по набору данных.
33. Вставка, добавление и удаление записей таблиц базы данных. Поиск, фильтрация и сортировка данных. Возможности среды для создания новых полей.
34. Принцип разработки программы, управляющей удаленной базой данных. SQL запросы на выборку, добавление, редактирование и удаление данных. Агрегатные функции. Группировка записей.
35. Статические и динамические параметрические SQL-запросы.

## **3.2. Типовые задания для оценки освоения междисциплинарного курса «МДК.01.02 Поддержка и тестирование программных модулей»**

### **3.2.1. Типовые задания в тестовой форме для оценки знаний**

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 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
  
4. 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
  
5. 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
  
6. 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

7. 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
8. Какие существуют методы построения тестовых путей?
- a. статические**
  - b. динамические**
  - c. методы реализуемых путей**
9. 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**
10. Какие существуют разновидности интеграционного тестирования?
- a. монолитное тестирование**
  - b. нисходящее тестирование**
  - c. восходящее тестирование**
  - d. Регрессионное тестирование
11. Каковы особенности нисходящего тестирования?
- a. необходимость разработки заглушек**
  - b. необходимость разработки среды управления очередностью вызовов модулей**
  - c. параллельная разработка эффективных модулей
12. 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
13. Возможно ли тестирование программы на всех допустимых значениях параметров?
- a. никогда
  - b. да, всегда
  - c. возможно в отдельных случаях**
14. 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»

15. 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

16. 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

17. 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

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

- a. нет
- b. да

19. 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

20. Какими свойствами обладает метод повторного прогона всех тестов?

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

21. 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

### 3.2.2 Задания для выполнения внеаудиторной самостоятельной работы

1. Дан двухмерный массив  $5 \times 5$ . Найти сумму модулей отрицательных нечетных элементов. Сформулировать требования к программному продукту и выполнить анализ и тестирование программных требований в соответствии со свойствами качественных требований.

2. Дана матрица. Вывести на экран все четные строки, то есть с четными номерами, у которых первый элемент больше последнего. Сформулировать требования к программному продукту и разработать чек – лист.

3. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту и разработать чек – лист.

4. Дана целочисленная квадратная матрица. Определить: произведение элементов в тех строках, которые не содержат отрицательных элементов. Сформулировать требования к программному продукту и разработать тест – кейс.

5. Для заданной матрицы размером 8 на 8 найти такие  $k$ , что  $k$ -я строка матрицы совпадает с  $k$ -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент. Сформулировать требования к программному продукту и разработать набор тест – кейсов.

6. Две строки матрицы назовем *похожими*, если совпадают множества чисел, встречающихся в этих строках. Найдите все пары похожих строк в заданной матрице  $m \times n$ . Сформулировать требования к программному продукту и разработать набор тест – кейсов.

7. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту. Выполнить тестирование программного продукта по структурным критериям.

8. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту. Выполнить тестирование программного продукта по методу «белого ящика».

9. Дана матрица. Вывести на экран все четные строки, то есть с четными номерами, у которых первый элемент больше последнего. Сформулировать требования к программному продукту и выполнить тестирование программного продукта по методу «белого ящика».
10. Для заданной матрицы размером 8 на 8 найти такие  $k$ , что  $k$ -я строка матрицы совпадает с  $k$ -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент. Сформулировать требования к программному продукту и выполнить тестирование по методу «белого ящика».
11. Для заданной матрицы размером 8 на 8 найти такие  $k$ , что  $k$ -я строка матрицы совпадает с  $k$ -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент. Сформулировать требования к программному продукту и выполнить тестирование программного продукта по структурным критериям.
12. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту. Выполнить тестирование программного продукта по функциональным критериям.
13. В матрице  $m \times n$ . Отсортировать по убыванию элементы строк, расположенные после второго отрицательного числа. Сформулировать требования к программному продукту. Выполнить тестирование программного продукта по методу «черного ящика».
14. Для заданной матрицы размером 8 на 8 найти такие  $k$ , что  $k$ -я строка матрицы совпадает с  $k$ -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент. Сформулировать требования к программному продукту и выполнить тестирование по методу «черного ящика».
15. Описать функцию  $NMin(A,N)$  и  $NMax(A,N)$  целого типа, находящую номер минимального и максимального элемента массива  $A$  (массив состоит из  $N$  вещественных чисел). Сформулировать требования к программному продукту и выполнить тестирование по методу «белого ящика».
16. Описать функцию  $NMin(A,N)$  и  $NMax(A,N)$  целого типа, находящую номер минимального и максимального элемента массива  $A$  (массив состоит из  $N$  вещественных чисел). Сформулировать требования к программному продукту и выполнить модульное тестирование
17. Описать функцию  $NewStr(S)$ , удаляющую в строке  $S$  начальные и конечные пробелы. В основной программе ввод строки, обращение методу - функции и вывод результата. Сформулировать требования к программному продукту и выполнить unit – тестирование.
18. Описать функцию  $NewStr(S)$ , удаляющую в строке  $S$  начальные и конечные пробелы. В основной программе ввод строки, обращение методу - функции и вывод результата. Предусмотреть использование 2 –х форм. Сформулировать требования к программному продукту и выполнить интеграционное тестирование.
19. Описать функцию  $NMin(A,N)$  и  $NMax(A,N)$  целого типа, находящую номер минимального и максимального элемента массива  $A$  (массив состоит из  $N$

вещественных чисел). Предусмотреть использование 2 –х форм. Сформулировать требования к программному продукту и выполнить интеграционное тестирование.

20. Дано натуральное число  $n$  и последовательность из 5 чисел. Найти количество чисел, являющихся степенями пятерки. Определить функцию пользователя, позволяющую распознавать степень пятерки. В основной программе ввод чисел, обращение к функции, вычисление количества и вывод результата. Сформулировать требования к программному продукту и выполнить модульное тестирование.

Темы рефератов (докладов, презентаций)

1. Принцип построения тестового набора данных и составления отладочных заданий.
2. Методы отладки программного продукта
3. Методы ручного тестирования
4. Метод обратного прослеживания
5. Инструментальные средства отладки ПП
6. Модели жизненного цикла программного средства.
7. Разработка программного продукта с использованием подпрограммы-процедуры.
8. Модульное программирование.
9. Осуществление разработки кода программного модуля на современных языках программирования
10. Реализация процедур и функций работы с бинарным деревом.
11. Объектно-ориентированное проектирование.
12. Основы языка UML (Unified Modeling Language).
13. Структура программы на языке C++. Проект.
14. Стандартная библиотека функций языка C++.
15. Компиляция программы и сборка исполняемого модуля.
16. Виртуальные функции и абстрактные базовые классы.
17. Множественное наследование.
18. Ассоциативные массивы.
19. Объекты-функции и предикаты.

### **3.2.3. Типовые задания для другой формы промежуточной аттестации по междисциплинарному курсу «МДК.01.02 Поддержка и тестирование программных модулей»**

1. Тестирование как часть процесса верификации программного обеспечения.
2. Методы тестирования.
3. Классификация тестирования по уровням.
4. Принципы тестирования и отладка программного обеспечения.
5. Цели и задачи тестирования программного обеспечения.
6. Этапы тестирования программного обеспечения.
7. Комплексное тестирование программного обеспечения.
8. Восходящее и нисходящее тестирование.
9. Стратегия тестирования и отладки программного обеспечения.
10. Метод Сандвича.

11. Метод «белого ящика».
12. Метод «черного ящика».
13. Метод отладки программного обеспечения.

1. Виды контроля качества разрабатываемого ПО.
2. Ручной контроль.
3. Структурное, функциональное и оценочное тестирование.
4. Классификация ошибок.
5. Методы и средства отладки ПО.

1. Функциональное тестирование.
2. Нефункциональное тестирование.
3. Тестирование производительности.
4. Прочие виды нефункционального тестирования.
5. Средства разработки технической документации.
6. Технологии разработки документов.
7. Документирование программного обеспечения в соответствии с Единой системой программной документации.
8. Автоматизация разработки технической документации.
9. Автоматизированные средства оформления документации.

### **3.2.4. Типовые задания для контрольной работы по междисциплинарному курсу «МДК.01.02 Поддержка и тестирование программных модулей»**

1. Дан массив  $A$  из  $n$  целых чисел. Найти сумму максимального и минимального элемента в массиве. (Поиск максимума и минимума реализовать с помощью подпрограмм-функций).
2. Дан файл целых чисел. Выбрать наибольшее из чисел, принадлежащее интервалу  $[a, b]$ . Концы интервала  $a$  и  $b$  вводятся с клавиатуры.
3. Дан текстовый файл  $F1$ . Переписать его содержимое в файл  $F2$ , сохраняя строчную структуру и удаляя пустые строки.
4. Даны две символьные строки  $S1$  и  $S2$ , содержащие только строчные латинские буквы. Построить строку  $S3$ , в которую войдут только общие символы  $S1$  и  $S2$  в алфавитном порядке и без повторений.
5. Дан файл целых чисел. Определить, сколько раз в нем повторяется максимальное значение.
6. Дан файл целых чисел. Определить, сколько раз в нем повторяется максимальное значение.
7. По координатам вершин треугольника вычислить его периметр, используя подпрограмму вычисления длины отрезка, соединяющего две точки. (длина отрезка =  $\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$ ), где  $(x_1, y_1)$  - координаты одной точки,  $(x_2, y_2)$  - координаты второй точки отрезка).
8. Даны два файла целых чисел. Определить, в каком из них больше положительных, отрицательных и нулевых значений.
9. Составить рекурсивную подпрограмму вычисления  $N!$
10. Дана вещественная матрица размера  $m \times n$ . Найти значение наибольшего по модулю элемента матрицы и указать его местоположение в матрице.



11. Определить среднее арифметическое чисел, хранящихся в файле Note.txt.
12. Дан список L, из N целых чисел. Удалить первое вхождение максимального элемента в списке.
13. Дан список L, из N целых чисел. Удалить первое вхождение минимального элемента в списке.
14. Дан текстовый файл Note.txt. Определить длину самой длинной строки этого файла.
15. Разработать и произвести отладку программы: Найти сумму бесконечного ряда. Суммировать до тех пор, пока сумма не станет больше заданного  $p > 0$ . Вывести эти числа.
16. Разработать и произвести отладку программы для определения  $N! - M!$ .  $N! = 1 * 2 * 3 * 4 * \dots * n$
17. Разработать и произвести отладку программы: Вычислить сумму квадратов всех целых чисел, пока сумма квадратов меньше заданного числа A. Вывести эти числа.
18. Разработать и произвести отладку программы: Произведение первых четных чисел равно P, сколько сомножителей взято.
19. Разработать и произвести отладку программы: Определить все двузначные числа, сумма квадратов цифр которых кратны числу 15.
20. Разработать и произвести отладку программы: Даны два одномерных массива одинаковой длины. Получить третий массив такой же размерности, каждый элемент которого равен сумме соответствующих элементов данных массивов.
21. Разработать и произвести отладку программы: дан одномерный массив чисел. Определите сумму элементов, принадлежащих промежутку от A до B (A и B вводить с клавиатуры).
22. Разработать и произвести отладку программы определения количества элементов массива, больших среднего арифметического всех его элементов.
23. Разработать и произвести отладку программы: ан массив P целых чисел из n элементов, заполненный случайным образом числами из промежутка [-30,30]. Из элементов массива P сформировать массив M из четных чисел. Исходный и скорректированный массивы вывести на экран.
24. Разработать и произвести отладку программы: ан массив P целых чисел из n элементов, заполненный случайным образом числами из промежутка [-10,10]. Из элементов массива P сформировать массив M той же размерности по возрастанию. Исходный и скорректированный массивы вывести на экран.
25. Разработать и произвести отладку программы, печатающей все делители целого числа в порядке возрастания
26. Разработать и произвести отладку программы для решения квадратного уравнения.
27. Создать и отладить приложение – конвертор перевода суммы денег из долларов в рубли.
28. Разработать и произвести отладку программы для вычисления делителей натурального числа N. Вывести сами делители, их количество.
29. Разработать и произвести отладку программы, вычисляющей сумму 1-й и последней цифр натурального числа N. Вывести эти цифры и сумму.

30. Создать и отладить приложение для решения квадратного уравнения.
31. Разработать и произвести отладку программы, находящей все простые числа в заданном диапазоне.
32. Разработать и произвести отладку программы, находящей все нечетные числа в заданном диапазоне и их количество.
33. Разработать и произвести отладку программы, находящей все четные числа в заданном диапазоне и их количество.
34. Разработать и произвести отладку программы, которая задает размер линейного массива, заполняет этот массив случайными целыми числами, выводит список элементов массива; отсортировать массив по убыванию.
35. Разработать и произвести отладку программы, которая задает размер линейного массива, заполняет этот массив случайными целыми числами, выводит список элементов массива; отсортировать массив по возрастанию
36. Разработать и произвести отладку программы, которая задает размер линейного массива, заполняет этот массив случайными целыми числами, выводит список элементов массива; поменять местами два элемента массива с номерами k1 и k2.
37. Разработать и произвести отладку программы, которая задает размер линейного массива, заполняет этот массив случайными целыми числами, выводит список элементов массива, определяет минимальный и максимальный элементы массива.
38. Разработать и произвести отладку программы, которая задает размер линейного массива, заполняет этот массив случайными целыми числами, выводит список элементов массива, определяет сумму всех элементов и количество положительных элементов.

### **3.3. Типовые задания для оценки освоения междисциплинарного курса «МДК.01.03 Разработка мобильных приложений»**

#### **3.3.1. Типовые задания в тестовой форме для оценки знаний**

1. Набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения называется:
  - а) Android SDK**
  - б) JDK
  - в) плагин ADT
  - г) Android NDK
2. С какой целью был создан Open Handset Alliance?
  - а) писать историю развития ОС Android
  - б) продавать смартфоны под управлением Android
  - в) рекламировать смартфоны под управлением Android
  - г) 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?
  - а) позволить разработчикам оптимизировать загрузенность системы при**

## использовании процедур OpenGL

- б) для ускорения работы эмулятора в среде разработки
- в) для оптимизированной обработки данных и изображений
- г) позволить разработчикам эффективно распараллелить C++ мобильные приложения

## 4. Библиотеки, реализованные на базе PacketVideo OpenCORE:

### а) Media Framework

- б) SQLite
- в) FreeType
- г) 3D библиотеки

## 5. Какой движок баз данных используется в ОС Android?

- а) InnoDB
- б) DBM
- в) MyISAM
- г) **SQLite**

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

- а) **для оптимизированной обработки данных и изображений**
- б) позволить разработчикам оптимизировать загрузенность системы при использовании процедур OpenGL
- в) для ускорения работы эмулятора в среде разработки
- г) позволить разработчикам эффективно распараллелить C++ мобильные приложения

## 7. Intel XDK поддерживает разработку под:

- а) JavaFX Mobile
- б) Apple iOS, BlackBerry OS
- в) MtkOS, Symbian OS, Microsoft Windows 8
- г) **Android, Apple iOS, Microsoft Windows 8, Tizen**

## 8. Каждый приемник широковещательных сообщений является наследником класса ...

- а) ViewReceiver
- б) IntentReceiver
- в) ContentProvider
- г) **BroadcastReceiver**

## 9. Какой класс является основным строительным блоком для компонентов пользовательского интерфейса (UI), определяет прямоугольную область экрана и отвечает за прорисовку и обработку событий?

- а) GUI
- б) **View**
- в) UIComponent

г) Widget

10. Какой слушатель используется для отслеживания события касания экрана устройства?

- а) OnPressListener
- б) OnTouchListener**
- в) OnClickListener
- г) OnInputListener

11. В какой папке необходимо разместить XML файлы, которые определяют все меню приложения?

- а) res/value
- б) res/items
- в) res/layout
- г) res/menu**

12. Фоновые приложения ...

- а) после настройки не предполагают взаимодействия с пользователем, большую часть времени находятся и работают в скрытом состоянии**
- б) выполняют свои функции и когда видимы на экране, и когда скрыты другими приложениями
- в) небольшие приложения, отображаемые в виде графического объекта на рабочем столе
- г) большую часть времени работают в фоновом режиме, однако допускают взаимодействие с пользователем и после настройки

13. Полный иерархический список обязательных файлов и папок проекта можно увидеть на вкладке ...

- а) Package Explorer**
- б) Internet Explorer
- в) Navigator
- г) Project Explorer**

14. Какой компонент управляет распределенным множеством данных приложения?

- а) сервис (Service)
- б) активность (Activity)
- в) приемник широковещательных сообщений (Broadcast Receiver)
- г) контент-провайдер (Content Provider)**

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

- а) html
- б) xml**
- в) gml
- г) xhtml

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

- а) Создание класса наследника от класса ContentProvider; Определение способа организации данных; Заполнение контент-провайдера данными
- б) Проектирование способа хранения данных; Определение способа организации данных;
- в) Создание класса наследника от класса ContentProvider; Заполнение контент-провайдера данными; Определение способа работы с данными
- г) **Проектирование способа хранения данных; Создание класса наследника от класса ContentProvider; Определение строки авторизации провайдера, URI для его строк и имен столбцов**

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

- а) **представляет собой структуру данных, содержащую описание операции, которая должна быть выполнена, и обычно используется для запуска активности или сервиса**
- б) используется для передачи сообщений пользователю
- в) используется для получения инструкций от пользователя
- г) **используются для передачи сообщений между основными компонентами приложений**

18. Расположение элементов мобильного приложения:

- а) полезно для передачи иерархии
- б) влияет на удобство использования
- в) полезно для создания пространственных отношений между объектами на экране и объектами реального мира
- г) **все варианты ответа верны**

19. Какие элементы управления применяются для действий по настройке?

- а) командные элементы управления
- б) **элементы выбора**
- в) элементы ввода
- г) элементы отображения

20. Примерами комбо-элементов не являются:

- а) комбо-список
- б) все вышеперечисленное
- в) комбо-кнопка
- г) **комбо-поле**

21. Дизайн или проектирование интерфейса для графических дизайнеров:

- а) **все варианты ответа верны**
- б) прозрачность и понятность информации
- в) тон, стиль, композиция, которые являются атрибутами бренда
- г) передача информации о поведении посредством ожидаемого назначения

22. Более крупные элементы:

- а) привлекают больше внимания**
- б) все варианты ответа верны
- в) размер не влияет на уровень внимания
- г) привлекают меньше внимания

23. К традиционным типографическим инструментам не относят

- а) масштаб
- б) цвет**
- в) разреженность
- г) выравнивание по сетке

24. К элементам ввода относят:

- а) ограничивающие элементы ввода
- б) ползунки
- в) счетчики
- г) все вышеперечисленное**

25. Выделяют следующие категории плотности экрана для Android-устройств:

- а) LDPI, XHDPI, XXHDPI, и XXXHDPI
- б) правильный вариант ответа отсутствует
- в) LDPI, MDPI, HDPI, XHDPI, XXHDPI, и XXXHDPI**
- г) LDPI, MDPI, HDPI

26. Следующие утверждения не верны:

- а) не используйте интерфейсные элемент**
- б) картинки работают быстрее, чем слова
- в) на любом шаге должна быть возможность вернуться назад
- г) если объекты похожи, они должны выполнять сходные действия

27. Следующие утверждения верны:

- а) текстура бесполезна для передачи различий или привлечения внимания
- б) восприятие направления затруднено при больших размерах объектов
- в) все варианты ответа верны**
- г) люди легко воспринимают контрастность

28. Основные вкладки (FixedTabs) удобны при отображении

- а) от четырех вкладок
- б) двух вкладок
- в) трех и более вкладок
- г) трех и менее вкладок**

29. Диалоговое окно, содержащее линейку процесса выполнения какого-то действия — это

- а) DatePickerDialog

- б) AlertDialog
- в) ProgressDialog**
- г) DialogFragment

30. Уведомления стоит использовать, когда

- а) сообщение не требует ответа пользователя, но важно для продолжения его работы
- б) сообщение является важным и требует немедленного прочтения и ответа
- в) сообщение является важным, однако требует немедленного прочтения, но не ответа
- г) сообщение является важным, однако не требует немедленного прочтения и ответа**

31. Какой метод запускает новую активность?

- а) startActivity()**
- б) beginActivity()
- в) intentActivity()
- г) newActivity()

### 3.3.2 Задания для выполнения внеаудиторной самостоятельной работы

1. Требуется разработать приложение с графическим пользовательским интерфейсом, поддерживающее создание/редактирование/удаление/поиск заметок. Два варианта хранения заметок: А) в базе SQLite. Б) С использованием файловой системы.
2. Создать приложение с графическим пользовательским интерфейсом с функциями: Определение местоположения пользователя на карте Google Map; Определение скорости и направления движения пользователя; Масштабирование карты. Программа должна быть конфигурируемой. Настройки: Режим определения местоположения (через GPS либо по сотам); Включение/отключение режима поиска.
3. Разработать приложение-таймер с использованием датчика ориентации в виде песочных часов. Каждый раз для того чтобы активировать таймер, необходимо перевернуть экран мобильного устройства вверх ногами. Используйте анимацию для показа «перетекающего песка» и переворота песочных часов. Для задания времени перетекания песка требуется разработать push-notification сервер. Через форму ввода на сервере можно отправлять на клиент (приложение-таймер) указанное время (числовой ввод).
4. Разработка программы для обмена мгновенными сообщениями. Требуется разработать приложение для обмена мгновенными сообщениями через Wi-Fi/Bluetooth. Поддерживаемые режимы: 1. Активный режим. Приложение занимает весь экран, содержит поля для отправки сообщений и список принятых сообщений. 2. Режим уведомлений. Приложение через уведомления показывает принятые сообщения.
5. Разработка мобильного сайта с адаптивным дизайном. Требуется разработать сайт, пригодный для просмотра на смартфонах и планшетах, с использованием принципов адаптивного дизайна: 1) Гибкая сетка. 2) Пропорциональные

шрифты. 3) Масштабируемые изображения. 4) Медиазапросы

Темы рефератов (докладов, презентаций)

1. Программный стек Android. Виртуальная машина Dalvik.
2. Архитектура Android-приложений.
3. Четыре пункта философии разработки приложений под Android.
4. Приемы для улучшения производительности и уменьшения потребления памяти для приложений Android.
5. Основные составляющие манифеста приложения.
6. Жизненный цикл мобильного приложения.
7. Разработка интерфейсов, не зависящих от разрешения и плотности пикселей.
8. Intents и Activities. Принципы работы Intent-фильтров.
9. Адаптеры и привязка данных.
10. Работа с интернет-ресурсами.
11. Диалоговые окна: создание и использование.
12. Курсоры, Content Values. Получение данных из SQLite.
13. Фоновые службы, toast-уведомления и сигнализация.
14. Геолокационные и картографические сервисы: конфигурирование и использование.
15. Сенсорные датчики. Sensor manager.
16. Анимация и спецэффекты.
17. Акселерометр, датчик ориентации и компас: регулировка и программные функции.
18. Межпроцессное взаимодействие. Язык AIDL.
19. Основные права и полномочия для запуска приложений на устройстве.
20. Работа с настройками сотовой сети, подключение голосовых услуг, получение и отправка коротких сообщений.

### **3.3.3. Типовые задания для другой формы контроля по междисциплинарному курсу «МДК.01.03 Разработка мобильных приложений»**

1. Классификация мобильных устройств, технические характеристики мобильных устройств. 2. Коммуникационные технологии, программные платформы.
3. Основные платформы мобильных приложений, сравнительная характеристика.
4. Нативные приложения.
5. Веб-приложения.
6. Гибридные и кроссплатформенные приложения, их области применения.
7. Основные языки для разработки мобильных приложений (Java, Objective-C и др.).
8. Инструменты разработки мобильных приложений (JDK/ AndroidStudio/ WebView/ Phonegap и др.).
9. Инструментарий среды разработки мобильных приложений.
10. Структура типичного мобильного приложения.
11. Элементы управления и контейнеры.
12. Работа со списками.
13. Способы хранения данных.



1. Инструменты для разработки и их установка: Java SDK.
2. Инструменты для разработки и их установка: android SDK.
3. Инструменты для разработки и их установка: Eclipse IDE for Java Developers
4. Создание эмулятора мобильного устройства,
5. Инструменты для разработки и их установка: ADT plugin.
6. Создание первого приложения.
7. Задание параметров интерфейса, единицы измерения.
8. Варианты Layout.
9. Инструменты для разработки и их установка.
10. Интерфейсные элементы.
11. Особенности среды разработки.
11. Отличия от android.

### **3.3.3. Типовые задания для дифференцированного зачета по междисциплинарному курсу «МДК.01.03 Разработка мобильных приложений»**

1. История возникновения мобильных операционных систем
2. Основные этапы становления рынка мобильных приложений
3. Современное состояние рынка мобильных приложений
4. Классификация видов мобильных приложений
5. Преимущества использования мобильных приложений в сравнении с вебприложениями
6. Недостатки использования мобильных приложений в сравнении с десктопными приложениями
7. Инструментальные среды разработки мобильных приложений для операционной системы Apple iOS
8. Инструментальные среды разработки мобильных приложений для операционной системы Android
9. Инструментальные среды разработки мобильных приложений для операционной системы Windows Phone
10. Структура операционной системы iOS
11. Структура операционной системы Android
12. Структура приложения iOS
13. Структура приложения Android
14. Основные требования к интерфейсу приложений iOS
15. Основные требования к интерфейсу приложений Android
16. Android-манифест
17. Взаимодействие Android-приложения с сетью
18. Работа Android-приложения с локальной базой данных
19. Считывание информации Android-приложением с XML-файла
20. Вызов приложения из другого приложения в ОС Android
21. Проблемы безопасности мобильных операционных систем
22. Бизнес-модели распространения мобильных приложений
23. Стратегия размещения приложения на Google Play
24. Стратегия размещения приложения на AppStore

25. Сравнительная характеристика современных мобильных операционных систем
26. HTML5 и мобильные приложения
27. Проблемы совместимости мобильных приложений со старыми версиями операционных систем
28. Проектирование мобильных приложений с использованием C++
29. Технологии фреймворков в проектировании мобильных приложений
30. Проблемы масштабирования СУБД в мобильных приложениях
31. Основные технологии виртуализации в инструментальных средах при создании мобильных приложений
32. Проблемы обеспечения безопасности в платных мобильных приложениях
33. Перспективы развития рынка мобильных приложений в России

### **3.4. Типовые задания для оценки освоения междисциплинарного курса «МДК.01.04 Системное программирование»**

#### **3.4.1. Типовые задания в тестовой форме для оценки знаний**

1. Операционная система:

1. **система программ, которая обеспечивает совместную работу всех устройств компьютера по обработке информации**
2. система математических операций для решения отдельных задач
3. система планового ремонта и технического обслуживания компьютерной техники

2. Программное обеспечение (ПО) – это:

1. **совокупность программ, позволяющих организовать решение задач на компьютере**
2. возможность обновления программ за счет бюджетных средств
3. список имеющихся в кабинете программ, заверен администрацией школы

3. Загрузка операционной системы – это:

1. запуск специальной программы, содержащей математические операции над числами
2. **загрузка комплекса программ, которые управляют работой компьютера и организуют диалог пользователя с компьютером**
3. вложение дискеты в дисковод

4. Система программирования – это:

1. комплекс любимых программ программиста
2. **комплекс программ, облегчающий работу программиста**
3. комплекс программ, обучающих начальным шагам программиста

5. Прикладное программное обеспечение – это:

1. справочное приложение к программам
2. **текстовый и графический редакторы, обучающие и тестирующие программы, игры**
3. набор игровых программ

6. Прикладное программное обеспечение:

1. программы для обеспечения работы других программ
2. **программы для решения конкретных задач обработки информации**
3. программы, обеспечивающие качество работы печатающих устройств

7. Операционные системы:

1. **DOS, Windows, Unix**
2. Word, Excel, Power Point
3. (состав отделения больницы): зав. отделением, 2 хирурга, 4 мед. Сестры

8. Системное программное обеспечение:

1. **программы для организации совместной работы устройств компьютера как единой системы**
2. программы для организации удобной системы размещения программ на диске
3. набор программ для работы устройства системного блока компьютера

9. Сервисные (обслуживающие) программы:

1. программы сервисных организаций по бухгалтерскому учету
2. программы обслуживающих организаций по ведению делопроизводства
3. **системные оболочки, утилиты, драйвера устройств, антивирусные и сетевые программы**

10. Системные оболочки – это:

1. специальная кассета для удобного размещения дискет с операционной системой
2. **специальная программа, упрощающая диалог пользователь – компьютер, выполняет команды операционной системы**
3. система приемов и способов работы конкретной программы при загрузке программ и завершении работы

11. Использование одного имени для задания общих для класса действий, что означает способность объектов выбирать внутренний метод, исходя из типа данных, определяет свойство ООП

1. Полиморфизм
2. Управление событиями
3. **Инкапсуляция**
4. Наследование

12. Понятие «инкапсуляция» относится к

1. Технологии модульного программирования
2. **Технологии объектно – ориентированного программирования**
3. Технологии императивного программирования
4. Технологии модульного программирования

13. Свойство ООП, которое может быть смоделировано с помощью таксономической классификационной схемы (иерархии) называется
1. Инкапсуляция
  2. Управление событиями
  3. Полиморфизм
  4. **Наследование**
14. Понятие класса в ООП включает в себя
1. **Поля и методы класса**
  2. Процедуры и функции обработки
  3. Поля и функции обработки
  4. Поля и процедуры обработки
15. Назначение конструктора объекта
1. **Только выделяет память под объект**
  2. Выделяет память и задает начальное значение полям
  3. Задает начальное значение полям
  4. Выделяет память, задает начальное значение полям, выполняет любые проверки, заданные программистом
16. Как описывается конструктор объекта
1. procedure create;
  2. constructor create;
  3. function create;
  4. **function constructor;**
17. Как описывается деструктор объекта
1. procedure free;
  2. **destructor free;**
  3. free;
  4. function free;
18. Понятия объекта в ООП - это
1. представитель класса
  2. **конкретные данные, заданные в классе.**
  3. компонент панели инструментов
  4. встроенный объект Delphi
19. Моделями типа «черный ящик» являются
1. Модели мышления
  2. Модели, описывающие зависимость параметров состояния объекта от входных параметров
  3. **Модели, описывающие входные и выходные параметры объекта без учета внутренней структуры объекта**
  4. Модели «аварийного» ящика на самолетах
20. Моделями типа «белый ящик» являются

1. Модели мышления
2. **Модели, описывающие зависимость параметров состояния объекта от входных параметров**
3. Модели, описывающие входные и выходные параметры объекта с учетом внутренней структуры объекта
4. Модели, описывающие выходные данные в программе

21. Программа «драйвер» служит для

1. запуска программы на выполнение
2. имитации запуска программы на выполнение
3. **проверки правильности работы программы**
4. передачи параметров в процедуры и функции

22. Программа «заглушка» служит для

1. запуска программы на выполнение
2. **имитации запуска другой программы на выполнение**
3. проверки правильности работы программы
4. имитации передачи параметров в другой модуль

23. Какие методы сборки программы существуют

1. монолитная
2. **пошаговая**
3. **одновременная**
4. постепенная

24. Какой метод тестирования программы учитывает закон распределения входных данных

1. детерминированное тестирование
2. функциональное тестирование
3. стохастическое тестирование
4. **логическое тестирование**

25. Программирование сверху вниз – это

1. Процесс, при котором от начального предположения осуществляется движение по направлению к лучшим решениям
2. **Процесс пошагового разбиения алгоритма на все более мелкие части с целью получения таких элементов, для которых можно написать конкретные команды**
3. Метод сведения трудной задачи к последовательности более простых
4. Исследование древовидной модели пространства решений и ориентация на поиск оптимального решения

26. Загрузочный модуль программы – результат работы

1. Грамматики
2. **Транслятора**
3. Интерпретатора
4. Редактора связей (компоновщика)

27. Интегрированная система программирования включает компонент для перевода исходного текста программы в машинный код, который называется

1. построителем кода
2. **компилятор**
3. переводчиком
4. преобразователем

28. Результатом компиляции программы на языке высокого уровня является

1. Командный файл
2. **Объектный файл**
3. Исходный текст программы на языке высокого уровня
4. Дисплейный файл

### **3.4.2 Задания для выполнения внеаудиторной самостоятельной работы**

1. Написать программу, которая удаляет из введенной с клавиатуры строки первые пробелы (если их нет, то оставить как есть)
2. Подсчитать количество слов в строке
3. Выделить первое слово из строки
4. Найти максимальное из трех введенных целых чисел
5. Найти сумму цифр трехзначного числа
6. Решить квадратное уравнение  $ax^2+bx+c=0$ , коэффициенты уравнения вводятся с клавиатуры
7. Определить, кратна ли трем сумма цифр трехзначного числа. Число вводится с клавиатуры
8. Найти площадь прямоугольного треугольника по известным катету и гипотенузе. Исходные данные вводятся с клавиатуры
9. Переписать все символы строки в обратном порядке
10. Написать программу для сортировки одномерных массивов
11. Вывести на экран все трехзначные числа, в которых хотя бы две цифры одинаковые
12. Вывести на экран все трехзначные числа, которые начинаются и заканчиваются на одну и ту же цифру
13. Вывести на экран таблицу значений функции  $y=x^3$  при  $x=-4...4$ . Шаг изменения  $x$  вводится с клавиатуры
14. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 найти сумму положительных элементов
15. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 количество отрицательных элементов
16. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 найти среднее арифметическое четных элементов
17. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 заменить все отрицательные элементы нулями
18. В одномерном массиве, состоящем из 10 случайных чисел от -10 до 10 переставить местами минимальный и максимальный элементы
19. Определить среднее арифметическое элементов главной диагонали

квадратной матрицы

20. Определить максимальный элемент побочной диагонали квадратной матрицы

21. Найти сумму нечетных элементов прямоугольного массива, состоящего из пяти строк и семи столбцов

22. В прямоугольном массиве, состоящем из 7 строк и 10 столбцов заменить все отрицательные элементы их абсолютными значениями

23. Определить, является ли сумма элементов матрицы 3x2 нечетной

24. В прямоугольном массиве, состоящем из 7 строк и 10 столбцов заменить все отрицательные элементы их абсолютными значениями

25. Определить, является ли сумма элементов матрицы 3x2 нечетной

#### Темы рефератов (докладов, презентаций)

1. Работа на ассемблере с портами ввода-вывода.

2. Визуальный анализ данных

3. Сравнительные характеристики автоматических дизассемблеров

4. Организация интерфейса с процедурой на ассемблере

5. Создание Windows – приложений на ассемблере

6. Программирование сопроцессора

7. Связь ассемблера с другими языками программирования (Pascal – ассемблер и C – ассемблер)

8. Сложные структуры данных в ассемблере (структуры, записи, списки, таблицы)

9. Работа с числами в формате BCD на ассемблере

10. Обработка цепочек элементов в ассемблере

11. Измерение скорости работы программы средствами профайлера

12. Построение лексического анализатора

13. Организация таблиц идентификаторов транслятора по методу бинарного дерева

14. Организация таблиц идентификаторов транслятора с использованием хеш-функций

15. Развитие отладчиков в современных системах программирования

16. Возможности Hex-редакторов

17. Преобразование чисел при вводе с консоли на ассемблере

18. Преобразование чисел при выводе на консоль на ассемблере

19. Управление цветом при выводе информации на консоль средствами ассемблера

20. Функции BIOS и MS DOS для работы с клавиатурой

21. Организация ввода-вывода в консольном приложении Windows

22. Программные средства обслуживания CD и DVD дисков

23. Сравнение средств разработки ассемблерных программ

24. Алгоритмы работы с памятью на уровне операционной системы (выделение, освобождение)

25. Использование концепции **.net** в системном программировании

26. Структура компьютерных технологий

27. Ошибки округления в арифметических командах

28. Использование флагов переноса и переполнения в командах целочисленной арифметики и передачи управления
29. Сравнительные характеристики трансляторов
30. Современные системы программирования
31. Организация серверных приложений (CORBA, COM/DCOM)

### **3.4.3. Типовые задания для экзамена по междисциплинарному курсу «МДК.01.04 Системное программирование»**

1. Подсистемы управления ресурсами.
2. Управление процессами.
3. Управление потоками.
4. Параллельная обработка потоков. Создание процессов и потоков.
5. Обмен данными между процессами. Передача сообщений.
6. Анонимные и именованные каналы.
7. Сетевое программирование сокетов.
8. Динамически подключаемые библиотеки DLL
9. Сервисы. Виртуальная память. Выделение памяти процессам.
10. Работа с буфером экрана.
11. Что представляет собой язык программирования Ассемблер.
12. Виды программного обеспечения: системное, прикладное и промежуточное (middleware) программное обеспечение.
13. Тенденции развития программного обеспечения
14. Разработка программного обеспечения. Этапы разработки программного обеспечения
15. Системное программирование – основные определения.
16. Команды языка ассемблера. Дизассемблирование кода. Анализ кода
17. Виды программного обеспечения
18. Работа с middleware.
19. Основные функции middleware
20. Работа с графическими интерфейсами
21. Содержательная постановка задачи
22. Разработка модели и выбор метода решения.
23. Кодирование алгоритма.
24. Компиляция программы.
25. Тестирование программы.
26. Сопровождение программы.
27. Создание документации.
28. Мнемоника языка Ассемблера
29. Операнды языка Ассемблера.
30. Комментарии языка Ассемблера.
31. Составление программы на языке Ассемблера

## **4. Требования к дифференцированному зачету по учебной и производственной практике**

Целью оценки производственной практики является оценка уровня



сформированности профессиональных компетенций и динамика освоения общих компетенций. Дифференцированный зачет по учебной практике проводится путем решения практической задачи.

Результаты дифференцированного зачета выставляются с учетом данных аттестационного листа, отражающего качество текущего контроля с указанием видов работ, выполненных студентом во время учебной практики, их объема, качества выполнения руководителем практики.

Дифференцированный зачет по производственной практике проводится по результатам решения практической задачи. Условием допуска к дифференцированному зачету является получение опыта по виду профессиональной деятельности, предусмотренной модулем. Работодатель оценивает в аттестационном листе и характеристике профессиональной деятельности студента на практике качество выполнения работ в соответствии с технологией и требованиями организации, в которой он проходил практику. Аттестационный лист и характеристика профессиональной деятельности студента в период производственной практики заверяются подписью работодателя и печатью организации по месту прохождения практики.

«Положительная оценка» выставляется, если во время практики:

1) обучающийся показал достаточный уровень знания правовых основ деятельности полиции, проявлял способности в организации повседневной работы;

2) своевременно и качественно выполнил мероприятия, предусмотренные рабочим графиком (индивидуальным планом);

3) в соответствии с содержанием практики овладел необходимыми знаниями, умениями и навыками;

4) у обучающегося в полном объеме сформировались компетенции, предусмотренные программой практики.

«Неудовлетворительно» выставляется, если во время практики:

1) обучающийся не освоил основные положения нормативных актов, регламентирующих организацию деятельности полиции;

2) рабочий график (индивидуальный план) проведения практики выполнил не качественно и не в полном объеме;

3) не овладел необходимыми знаниями, умениями и навыками;

4) у обучающегося не сформировались в полном объеме компетенции, предусмотренные программой практики.

#### **4.1 Типовые задания для оценки умений и практического опыта по учебной практике**

1. Ознакомление с планом прохождения практики.
2. Инструктаж о прохождении практики.
3. Анализ нормативно-правовых актов и литературы.
4. Ознакомление с тематикой индивидуальных заданий для написания отчета.
5. Подготовка и оформление дневника
6. Приобретение навыков в разработке кода программного продукта на основе готовой спецификации на уровне модуля.
7. Использование инструментальных средств на этапе отладки программного

продукта.

8. Проведение тестирования программного модуля по определенному сценарию.
9. Использование инструментальных средств на этапе отладки программного продукта.
10. Разработка мобильных приложений.
11. Защита отчета по практике.

#### **4.2 Типовые задания для оценки освоения профессиональных и общих компетенций по производственной практике**

1. Ознакомление с планом прохождения практики.
2. Инструктаж о прохождении практики.
3. Анализ нормативно-правовых актов и литературы.
4. Ознакомление с тематикой индивидуальных заданий для написания отчета.
5. Подготовка и оформление дневника.
6. Приобретение навыков в разработке кода программного продукта на основе готовой спецификации на уровне модуля.
7. Использование инструментальных средств на этапе отладки программного продукта.
8. Проведение тестирования программного модуля по определенному сценарию.
9. Использование инструментальных средств на этапе отладки программного продукта.
10. Разработка мобильных приложений.
11. Защита отчета по практике.

### **5. Структура контрольно-оценочных материалов для экзамена по модулю**

Формой промежуточной аттестации по профессиональному модулю «ПМ.01 Разработка модулей программного обеспечения для компьютерных систем» является экзамен по модулю, на котором оценка освоения вида профессиональной деятельности, профессиональных компетенций осуществляется внешним экспертом от работодателя

Условием допуска студента к экзамену по модулю является представление аттестационных листов с положительной оценкой прохождения учебной и производственной практик, характеристики профессиональной деятельности студента от работодателя.

Разработанные задания к экзамену по модулю ориентированы на проверку освоения вида профессиональной деятельности в целом.

Экзамен по модулю предусматривает решение комплексной профессиональной задачи, имеющей критерии оценки и эталоны ответов. Каждое задание профессиональной задачи предусматривает оценку освоения определенной профессиональной компетенции.

Для эффективного выполнения заданий рабочее место студента оснащено необходимыми для работы программами: СПС «Консультант Плюс». Результаты выполнения заданий отражаются в листе «Отметка о

выполнении/невыполнении этапов задания» (Приложение 4). Итоговая оценка выставляется на основе критериев.

Положительная итоговая оценка является основанием для вынесения решения: вид профессиональной деятельности освоен», отрицательная оценка - «вид профессиональной деятельности не освоен».

Итоговый результат экзамена по модулю регистрируется в оценочной ведомости, оформляемой на каждого студента.

## I. ПАСПОРТ

Назначение:

Контрольно-оценочные материалы предназначены для контроля и оценки результатов освоения профессионального модуля «ПМ.01 Разработка модулей программного обеспечения для компьютерных систем» по специальности СПО базовый уровень подготовки код специальности 09.02.07 «Информационные системы и программирование».

Профессиональные компетенции:

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 1.4. Выполнять тестирование программных модулей.

ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.

ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.

Общие компетенции:

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.

ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.

ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения.

ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, 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. Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языках.

ОК 11. Использовать знания по финансовой грамотности, планировать предпринимательскую деятельность в профессиональной сфере.

Иметь практический опыт

- 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;
- выполнять оптимизацию и рефакторинг программного кода;
- оформлять документацию на программные средства.

Знать

- основные этапы разработки программного обеспечения;
- основные принципы технологии структурного и объектно-ориентированного программирования;
- способы оптимизации и приемы рефакторинга;
- основные принципы отладки и тестирования программных продуктов.

## II. ЗАДАНИЕ ДЛЯ ЭКЗАМЕНУЮЩЕГОСЯ

Инструкция. Внимательно прочитайте вопросы и условие практического задания экзаменационного билета. На основе исходных данных выполните задание и дайте правильные ответы на вопросы. Время выполнения заданий – 2 часа.

## III. ПАКЕТ ЭКЗАМЕНАТОРА

### IIIa. УСЛОВИЯ

Экзамен предполагает выполнение трех заданий – двух устных вопросов и одного практического задания. Экзамен проводится одновременно для всей учебной группы. Для эффективного выполнения заданий рабочее место студента оснащено необходимыми для работы программами: СПС «Консультант Плюс».

## Перечень практических заданий для экзамена по модулю

- Задание 1. Использование окон ввода. Дано двузначное число. Вывести вначале его левую цифру (десятки), а затем - его правую цифру (единицы).
- Задание 2. С начала суток прошло  $N$  секунд ( $N$  — целое). Найти количество полных минут, прошедших с начала суток.
- Задание 3. Дано трехзначное число. Проверить истинность высказывания: «Все цифры данного числа различны».
- Задание 4. Дана непустая строка. Вывести коды ее первого и последнего символа. Даны строки  $S$  и  $S_0$ . Найти количество вхождений строки  $S_0$  в строку  $S$ .
- Задание 5. Дано натуральное число  $n$ , массивы  $A$  и  $B$  размерности  $n$ . Вести с клавиатуры элементы массивов и вычислить сумму элементов массива  $A$ , количество положительных элементов массива  $B$  и его максимальный элемент.
- Задание 6. Дан массив  $A$  размерностью 10. Удалить минимальный элемент массива.
- Задание 7. Дана действительная матрица размера  $m \times n$ . Определить числа  $b_1, \dots, b_m$ , равные соответственно суммам положительных элементов строк. Отсортировать массив  $b$  по возрастанию.
- Задание 8. Дана целочисленная квадратная матрица порядка  $n$ . Найти номера строк, все элементы которых меньше по модулю первого.
- Задание 9. Дан массив данных об учениках класса: Фамилия, имя, отчество, адрес (улица, дом, квартира) и домашний телефон (если есть). Вывести на экран фамилию, имя и адрес тех учеников, до которых нельзя дозвониться.
- Задание 10. Протестировать тестовую базу данных методом тестирования программного обеспечения «черный ящик».
- Задание 11. Протестировать тестовую базу данных методом тестирования программного обеспечения «белый ящик».
- Задание 12. Протестировать тестовую базу данных модульным методом тестирования программного обеспечения.
- Задание 13. Протестировать тестовую базу данных интегрированным методом тестирования программного обеспечения.
- Задание 14. Создание проекта приложения в Android Studio. Создание приложения с одним экраном.
- Задание 15. Разработка мобильного приложения из четырех окон средствами Android Studio.
- Задание 16. Разработка программы «Будильник» для ОС Android средствами Android Studio.
- Задание 17. Разработка программы «Калькулятор» для ОС Android средствами Android Studio.
- Задание 18. Создание простых программ на языке Ассемблер.
- Задание 19. Создание программ с разветвлением на Ассемблер.
- Задание 20. Создание программ с использованием циклов на Ассемблер.
- Задание 21. Создание программ с использованием логических операций на Ассемблер.
- Задание 22. Создание программ с использованием циклов и логических операций на Ассемблер.

Задание 23. Создание программ с использованием команд сдвигов. Составление программ с использованием ввода-вывода на экран.

### **Перечень вопросов для экзамена по модулю**

1. Понятие ЖЦ ПО. Перечислите этапы ЖЦ ПО.
2. Технология структурного программирования.
3. Инструментальные средства оформления и документирования алгоритмов программ.
4. Оценка сложности алгоритма: классификация, классы алгоритмов, неразрешимые задачи.
5. Основные принципы объектно-ориентированного программирования. В чем его отличие от остальных видов программирования. Классы.
6. Перегрузка методов. Операции класса. Иерархия классов.
7. Синтаксис интерфейсов.
8. Интерфейсы и наследование. Структуры. Делегаты.
9. Регулярные выражения. Коллекции. Параметризованные классы. Указатели.
10. Операции со списками. Назначение и виды паттернов.
11. Основные шаблоны. Порождающие шаблоны. Структурные шаблоны. Поведенческие шаблоны.
12. Событийно-управляемое программирование.
13. Элементы управления. Диалоговые окна. Обработчики событий.
14. Введение в графику. Методы оптимизации программного кода.
15. Цели и методы рефакторинга.
16. Правила разработки интерфейсов пользователя.
17. Работа с базами данных. Доступ к данным.
18. Создание таблицы, работа с записями. Способы создания команд.
19. Тестирование как часть процесса верификации программного обеспечения.
20. Виды ошибок и методы отладки. Методы тестирования.
21. Классификация тестирования по уровням.
22. Тестирование производительности.
23. Регрессионное тестирование.
24. Средства разработки технической документации. Технологии разработки документов.
25. Документирование программного обеспечения в соответствии с Единой системой программной документации: основное.
26. Автоматизация разработки технической документации.
27. Автоматизированные средства оформления документации.
28. Основные платформы мобильных приложений, их сравнительная характеристику.
29. Нативные приложения, веб-приложения, гибридные и кроссплатформенные приложения, их области применения.
30. Перечислите основные языки для разработки мобильных приложений (Java, Objective-C и др.).
31. Инструменты разработки мобильных приложений (JDK/ AndroidStudio/ WebView/ Phonegap и др.)
32. Инструментарий среды разработки мобильных приложений.

33. Структура типичного мобильного приложения.
34. Элементы управления и контейнеры.
35. Работа со списками.
36. Способы хранения данных.
37. Подсистемы управления ресурсами.
38. Управление процессами. Управление потоками.
39. Параллельная обработка потоков.
40. Создание процессов и потоков.
41. Обмен данными между процессами. Передача сообщений.
42. Анонимные и именованные каналы.
43. Сетевое программирование сокетов.
44. Динамически подключаемые библиотеки DLL
45. Сервисы. Виртуальная память. Выделение памяти процессам.
46. Работа с буфером экрана.

### ШБ. КРИТЕРИИ ОЦЕНКИ

#### Критерии оценивания заданий

5 «отлично» - глубоко и прочно усвоен весь программный материал по модулю; последовательно и точно построена речь; теория увязывается с практической и профессиональной деятельностью; отсутствуют затруднения с ответами на дополнительные или уточняющие вопросы;

4 «хорошо» - усвоен весь программный материал; в речи имеются незначительные неточности; правильно применены теоретические знания; на большинство дополнительных или уточняющих вопросов дан ответ;

3 «удовлетворительно» - усвоена основная часть программного материала; речь не содержит «деталей»; недостаточно-правильные формулировки; затруднения в выполнении практических заданий; на большинство дополнительных или уточняющих вопросов испытываются затруднения в ответе;

2 «неудовлетворительно» - не усвоена значительная часть программного материала; ответ содержит существенные ошибки; затруднения в выполнении практических заданий, в формулировании основных дефиниций по курсу.

### 6. Учебно-методическое и информационное обеспечение модуля

#### *Основная литература:*

1. Гниденко, И. Г. Технология разработки программного обеспечения : учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2023. — 235 с. — (Профессиональное образование). — ISBN 978-5-534-05047-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/514591>
2. Кудрявцева, И. А. Программирование: комбинаторная логика : учебное пособие для среднего профессионального образования / И. А. Кудрявцева, М. В. Швецкий. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2022. — 524 с. — (Профессиональное образование). — ISBN 978-5-534-15128-2.

— Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/495225>

3. Кудрявцева, И. А. Программирование: теория типов : учебное пособие для среднего профессионального образования / И. А. Кудрявцева, М. В. Швецкий. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2022. — 652 с. — (Профессиональное образование). — ISBN 978-5-534-15382-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/498766>

4. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук ; под общей редакцией Д. В. Чистова. — Москва : Издательство Юрайт, 2023. — 258 с. — (Профессиональное образование). — ISBN 978-5-534-03173-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/513630>

5. Соколова, В. В. Разработка мобильных приложений : учебное пособие для среднего профессионального образования / В. В. Соколова. — Москва : Издательство Юрайт, 2023. — 175 с. — (Профессиональное образование). — ISBN 978-5-534-10680-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/518008>

#### ***Дополнительная литература:***

1. Гостев, И. М. Операционные системы : учебник и практикум для среднего профессионального образования / И. М. Гостев. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 164 с. — (Профессиональное образование). — ISBN 978-5-534-04951-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/514426>

2. Казарин, О. В. Основы информационной безопасности: надежность и безопасность программного обеспечения : учебное пособие для среднего профессионального образования / О. В. Казарин, И. Б. Шубинский. — Москва : Издательство Юрайт, 2023. — 342 с. — (Профессиональное образование). — ISBN 978-5-534-10671-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/518005>

3. Казарин, О. В. Программно-аппаратные средства защиты информации. Защита программного обеспечения : учебник и практикум для среднего профессионального образования / О. В. Казарин, А. С. Забабурин. — Москва : Издательство Юрайт, 2023. — 312 с. — (Профессиональное образование). — ISBN 978-5-534-13221-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/519364>

4. Нестеров, С. А. Базы данных : учебник и практикум для среднего профессионального образования / С. А. Нестеров. — Москва : Издательство Юрайт, 2023. — 230 с. — (Профессиональное образование). — ISBN 978-5-534-11629-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/518507>

5. Чернышев, С. А. Основы программирования на Python : учебное пособие для среднего профессионального образования / С. А. Чернышев. — Москва : Издательство Юрайт, 2023. — 286 с. — (Профессиональное образование). — ISBN 978-5-534-15160-2. — Текст : электронный // Образовательная платформа



Юрайт [сайт]. — URL: <https://urait.ru/bcode/519953>

**Интернет-источники:**

1. Огромный выбор конспектов лабораторных и практических работ, инструкционные и технологические карты почти на все основные темы курса <http://www.metod-kopilka.ru/page-2-1.html>
2. Электронная библиотечная система (ЭБС) «Университетская библиотека ONLINE» <https://urait.ru/>
3. Компьютерная справочная правовая система.

Приложение 1



**Частное профессиональное образовательное учреждение  
«Московский областной современный колледж»**

**ЗАЧЕТНО-ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ**

**20\_\_ / 20\_\_ учебный год**

Специальность:

Группа: \_\_\_\_\_ курс \_\_\_\_ семестр \_\_\_\_\_

Форма контроля – зачет, дифференцированный зачет, экзамен, другие формы контроля  
(подчеркнуть)

Наименование дисциплины:

Фамилия и инициалы преподавателя:

Дата проведения: «\_\_\_\_\_» \_\_\_\_\_ 20\_\_ г.

№ п/п	Фамилия, имя, отчество студентов	Номер зачетной книжки	Отметка о сдаче зачета / дифф. зачета / экзаменационной оценки / других формах контроля	Подпись преподавателя
1.				
2.				
3.				
4.				
5.				
6.				
7.				
8.				
9.				
10.				

Число студентов, явившихся на экзамен / зачет / дифф. зачет / другие формы контроля \_\_\_\_.  
Число студентов, не допущенных к экзамену / зачету / дифф. зачету / другим формам контроля \_\_\_\_.

Число студентов, не явившихся на экзамен / зачет / дифф. зачет / другие формы контроля \_\_\_\_.

Из них получивших «отлично» - \_\_\_\_, «хорошо» - \_\_\_\_, «удовлетворительно» - \_\_\_\_, «неудовлетворительно» - \_\_\_\_.

Из них получивших «зачтено» - \_\_\_\_, «не зачтено» - \_\_\_\_.

Преподаватель \_\_\_\_\_  
Подпись \_\_\_\_\_ Ф.И.О. \_\_\_\_\_

## Приложение 2



### Частное профессиональное образовательное учреждение «Московский областной современный колледж»

#### Сводная ведомость результатов освоения междисциплинарного курса

« \_\_\_\_ » \_\_\_\_\_ 202 \_\_\_\_ г.

Группа \_\_\_\_\_

Код, специальность: 09.02.07 Информационные системы и программирование

Дисциплина:

Преподаватель:

№ п/п	Фамилия, имя, отчество студентов	Оценка за выполнение внеаудиторных самостоятельных работ	Оценка за решение практической задачи	Оценка выполнения заданий в тестовой форме	Итоговая оценка по МДК
1.					
2.					
3.					
4.					
5.					
6.					
7.					
8.					
9.					

Преподаватель \_\_\_\_\_  
Подпись \_\_\_\_\_ Ф.И.О. \_\_\_\_\_





Частное профессиональное образовательное учреждение  
«Московский областной современный колледж»

**ЗАЧЕТНО-ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ**  
**2023 / 2024 учебный год**

Специальность: 09.02.07 Информационные системы и программирование

Группа \_\_\_ курс \_\_\_ семестр \_\_\_

Форма контроля – зачет, дифференцированный зачет, экзамен, другие формы контроля (подчеркнуть)

Наименование дисциплины: **Учебная практика (Разработка модулей программного обеспечения для компьютерных систем)**

Фамилия и инициалы преподавателя:

Дата проведения: «\_\_\_» \_\_\_\_\_ 20\_\_\_ г.

№ п/п	Фамилия, имя, отчество студентов	Номер зачетной книжки	Отметка о сдаче зачета / дифф. зачета / экзаменационной оценки / других формах контроля	Подпись преподавателя
1.				
2.				
3.				
4.				
5.				
6.				
7.				
8.				
9.				

Число студентов, явившихся на экзамен / зачет / дифф. зачет / другие формы контроля \_\_\_\_.

Число студентов, не допущенных к экзамену / зачету / дифф. зачету / другим формам контроля \_\_\_\_.

Число студентов, не явившихся на экзамен / зачет / дифф. зачет / другие формы контроля \_\_\_\_.

Из них получивших «отлично» - \_\_\_\_, «хорошо» - \_\_\_\_, «удовлетворительно» - \_\_\_\_, «неудовлетворительно» - \_\_\_\_.

Из них получивших «зачтено» - \_\_\_\_, «не зачтено» - \_\_\_\_.

Преподаватель \_\_\_\_\_

Подпись

\_\_\_\_\_

Ф.И.О.

**ОЦЕНОЧНАЯ ВЕДОМОСТЬ**

по профессиональному модулю

ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

от «15» января 2021 г.

Иванова Наталья Петрова обучающийся(-щаяся) по специальности 09.02.07 Информационные системы и программирование и модулю ПМ.01 Разработка модулей программного обеспечения для компьютерных систем в объеме 302 часа.

Результаты промежуточной аттестации по элементам профессионального модуля:

Элементы модуля (код и наименование МДК, код практик)	Формы промежуточной аттестации	Оценка
МДК.01.01 Разработка программных модулей	Другая форма контроля, защита курсовой работы, экзамен	
МДК.01.02 Поддержка и тестирование программных модулей	Другая форма контроля	
МДК.01.03 Разработка мобильных приложений	Другая форма контроля, Дифференцированный зачет	
МДК.01.04 Системное программирование	Дифференцированный зачет	
УП.01.01. Учебная практика	Дифференцированный зачет	
ПП.01.01. Производственная практика	Дифференцированный зачет	

Итоги экзамена по модулю по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

№ варианта задания	Коды проверяемых компетенций	Показатели оценки сформированности ПК	Оценка сформированности ПК	Оценка результатов освоения ПМ 03.
№ 1	ПК 1.1	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием	сформирована	Освоен
	ПК 1.2	Разрабатывать программные модули в соответствии с техническим заданием	сформирована	
	ПК 1.3	Выполнять отладку программных модулей с использованием специализированных программных средств	сформирована	
	ПК 1.4	Выполнять тестирование программных модулей	сформирована	
	ПК 1.5	Осуществлять рефакторинг и оптимизацию программного кода	сформирована	

Итоговая оценка

Председатель комиссии: \_\_\_\_\_

Члены комиссии: \_\_\_\_\_

**Лист отметки о выполнении/невыполнении заданий профессиональной задачи**  
 студентов гр. № \_\_\_\_\_ специальности 09.02.07 Информационные системы и программирование  
 по оценке результатов экзамена (квалификационного) по профессиональному модулю  
 ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

ФИО	№№ заданий, ПК и отметка о результате выполнения заданий (16 или 06)																		ИТОГО баллов	Результат: ПМ освоен / не освоен																				
	№1, ПК		№2, ПК		№3, ПК		№1, ПК		№2, ПК		№3, ПК		№1, ПК		№2, ПК		№3, ПК																							
	1.1		1.1		1.1		1.2		1.2		1.2		1.3		1.3		1.3				1.4		1.4		1.4		1.5		1.5		1.5									
	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0			1	0	1	0	1	0	1	0	1	0	1	0								
1.																																								
2.																																								
3.																																								
4.																																								
5.																																								
6.																																								
7.																																								
8.																																								
9.																																								
10.																																								
11.																																								
12.																																								
13.																																								
14.																																								
15.																																								
16.																																								
17.																																								
18.																																								
19.																																								
20.																																								

Критерии оценки результатов выполнения заданий:

- **оценка три** выставляется при условии правильности выполнения не менее семи заданий профессиональной задачи.
- **оценка четыре** выставляется при условии правильности выполнения не менее восьми заданий профессиональной задачи.
- **оценка пять** выставляется при условии правильности выполнения не менее десяти заданий профессиональной задачи.

Члены комиссии:

---



---