МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский ядерный университет «МИФИ»
Новоуральский технологический институт
филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ»
(НТИ НИЯУ МИФИ)
Колледж НТИ

Цикловая методическая комиссия информационных технологий

ОДОБРЕНО
Учёным Советом НТИ НИЯУ МИФИ
Протокол № 2 от 05 февраля 2024 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ ПМ.01 «РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ»

для студентов колледжа НТИ НИЯУ МИФИ,
обучающихся по программе среднего профессионального образования

специальность 09.02.07
«Информационные системы и программирование»

очная форма обучения
на базе основного общего образования

квалификация
программист

Новоуральск 2024

РАССМОТРЕНО:

на заседании

цикловой методической комиссии

информационных технологий

Протокол № 2 от 02.02.2024 г.

Председатель ЦМК ИТ

_____ И.И. Горницкая

Разработана на основе ФГОС СПО (утвержден Приказом Министерства образования и науки Российской Федерации от 09 декабря 2016 г. № 1547, зарегистрирован Министерством юстиции Российской Федерации 26 декабря 2016 г., регистрационный № 44936), с учетом примерной основной образовательной программы, в соответствии с действующим учебным планом, компетентностной моделью выпускника по специальности 09.02.07 Информационные системы и программирование

Фонд оценочных средств по профессиональному модулю ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем» – Новоуральск: Изд-во колледжа НТИ НИЯУ МИФИ, 2024. – 42с.

АННОТАЦИЯ

Фонд оценочных средств предназначен для текущего контроля и промежуточной аттестации обучающихся по специальности 09.02.07 Информационные системы и программирование на соответствие их персональных достижений поэтапным требованиям программы подготовки специалистов среднего звена по профессиональному модулю ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем». Комплектация фонда оценочных средств: паспорт, программа оценивания, оценочные средства для текущего контроля и промежуточной аттестации по профессиональному модулю, критерии оценивания. В паспорте фонда оценочных средств указаны: место профессионального модуля в структуре программы подготовки специалистов среднего звена, требования ФГОС СПО к результатам освоения профессионального модуля, перечень формируемых компетенций, компоненты фонда оценочных средств.

Разработчики: Горницкая И.И., преподаватель высшей категории, председатель ЦМК информационных технологий;

Лебедева А.Н., преподаватель первой категории ЦМК информационных технологий.

Тарасова А.В., преподаватель ЦМК информационных технологий.

Редактор: Горницкая И.И., преподаватель высшей категории, председатель ЦМК информационных технологий

СОДЕРЖАНИЕ

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ ПМ.01 «РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ» 4

2. ПРОГРАММА ОЦЕНИВАНИЯ МОДУЛЯ ПМ.01 «РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ» 7

3. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ 8

3.1 Тестовая форма контроля 9

3.2 Критерии оценки результатов выполнения тестовых заданий 12

3.3 Письменная форма контроля 12

3.4. Критерии оценки результатов выполнения письменных заданий 14

3.5 Устная форма контроля 14

3.6 Критерии оценки результатов выполнения устных заданий 19

4. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ 20

4.1. Критерии оценивания знаний, обучающихся на семестровом зачете 23

4.2 Критерии оценивания знаний, обучающихся на дифференцированном зачете 24

4.3 Критерии оценивания знаний, обучающихся на экзамене 24

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ ПМ.01 «РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ»

Фонд оценочных средств является составной частью учебно-методических документов, обеспечивающих реализацию основной профессиональной образовательной программы СПО по специальности 09.02.07 Информационные системы и программирование.

Фонд оценочных средств предназначен для проверки результатов освоения профессионального модуля ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем».

Место профессионального модуля в структуре основной профессиональной образовательной программы: профессиональный модуль ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем» принадлежит к профессиональному циклу.

Цель и планируемые результаты освоения профессионального модуля:

В результате изучения профессионального модуля студент должен освоить основной вид деятельности «Разработка модулей программного обеспечения для компьютерных систем» и соответствующие ему профессиональные компетенции, и общие компетенции:

Перечень формируемых компетенций в соответствии с требованиями ФГОС СПО:

Общие компетенции.

Код	Наименование общих компетенций
ОК 1	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.
ОК 2	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.
ОК 3	Планировать и реализовывать собственное профессиональное и

	личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях
ОК 4	Эффективно взаимодействовать и работать в коллективе и команде.
ОК 5	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.
ОК 6	Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения.
ОК 7	Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях.
ОК 8	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.
ОК 9	Пользоваться профессиональной документацией на государственном и иностранном языках.

Профессиональные компетенции

Код	**Наименование видов деятельности и профессиональных компетенций**
ВД 1	Разработка модулей программного обеспечения для компьютерных систем
ПК 1.1	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
ПК 1.2	Разрабатывать программные модули в соответствии с техническим заданием
ПК 1.3	Выполнять отладку программных модулей с использованием специализированных программных средств
ПК 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

В результате освоения профессионального модуля студент должен:

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

2. ПРОГРАММА ОЦЕНИВАНИЯ МОДУЛЯ ПМ.01 «РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ»

№ п/п	Контролируемые элементы профессионального модуля	Контролируемые компетенции	Форма аттестации
1	**2**	**3**	**4**
1	МДК 01.01. Разработка программных модулей	ОК 1 – ОК 9 ПК 1.1, ПК 1.2	IV семестр семестровый зачет
			V семестр дифференцированный зачет
			VI семестр комплексный экзамен
2	МДК 01.02 Поддержка и тестирование программных модулей	ОК 1 – ОК 9 ПК 1.3, ПК 1.4, ПК 1.5	V семестр семестровый зачет
			VI семестр комплексный экзамен
3	МДК 01.03. Разработка мобильных приложений	ОК 1 – ОК 9 ПК 1.2, ПК 1.6	VI семестр курсовой проект экзамен
4	МДК 01.04 Системное программирование	ОК 1 – ОК 9 ПК 1.2, ПК 1.3	VII семестр экзамен
5	УП 01.01 Учебная практика	ОК 1 – ОК 9 ПК 1.1-1.6	VI семестр дифференцированный зачет
6	ПП 01.01 Производственная практика	ПК 1.2-1.6	VI семестр дифференцированный зачет VII семестр дифференцированный зачет
7	ПМ 01. Разработка модулей программного обеспечения для компьютерных систем	ОК 1 – ОК 9 ПК 1.1-1.6	VII семестр экзамен по модулю

3. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ

Текущий контроль знаний, умений и навыков осуществляется в ходе образовательного процесса по всем изучаемым междисциплинарным курсам в формах, определяемых преподавателем в рабочей программе профессионального модуля. Он может проводиться в ходе всех видов занятий в форме, избранной преподавателем.

Текущий контроль успеваемости может проводиться в следующих формах:

– устная (устный опрос, защита письменной работы, доклад или реферат по результатам самостоятельной работы и т. д.);

– письменная (письменный опрос, выполнение контрольных и других проверочных работ и т. д.);

– тестовая (устное, письменное, компьютерное тестирование).

Возможны и другие виды текущего контроля успеваемости, которые определяются преподавателями.

ФОРМА КОНТРОЛЯ	МДК 01.01	МДК 01.02	МДК.01.03	МДК 01.04
тестовая	тема 1 тема 3 тема 5 тема 7	тема 1	тема 2	тема 1
письменная	тема 1-8	тема 1-2	тема 1-2	тема 1
устная	тема 1-8	тема 1-2	тема 1-2	тема 1

Система текущего контроля успеваемости предусматривает разнообразные по форме и содержанию контрольные мероприятия, учитывающие все виды аудиторной и самостоятельной учебной деятельности обучающегося.

Данные текущего контроля используются преподавателями для анализа (мониторинга) качества освоения обучающимися основной образовательной программы среднего образования, для организации консультаций.

3.1 Тестовая форма контроля

МДК.01.01. Разработка программных модулей

Тема 1. Жизненный цикл программного обеспечения.

Гниденко, И.Г. Технология разработки программного обеспечения: учебное пособие для среднего профессионального образования/ И.Г.Гниденко, Ф.Ф.Павлов, Д.Ю.Федоров.— 2-е изд., перераб. и доп.— Москва: Издательство Юрайт, 2023.— 248с

https://urait.ru/quiz/run-test/D0FE194F-CE4D-4581-9146-F734CB060787/4E12266E-B9A8-44A7-853C-80E46FACB923

Тема 3. Объектно-ориентированное программирование

Федоров, Д.Ю. Программирование на языке высокого уровня Python: учебное пособие для среднего профессионального образования/ Д.Ю.Федоров.— 5-е изд., перераб. и доп.— Москва: Издательство Юрайт, 2023.— 227с.

https://urait.ru/quiz/run-test/A1C48662-B22F-4B93-B641-2B7ABD63954F/ED760669-5483-4829-B94B-F0C55FE9BAC5

Тема 5. Событийно-управляемое программирование

Казанский, А.А.Программирование на Visual C#: учебное пособие для среднего профессионального образования/ А.А.Казанский.— 2-е изд., перераб. и доп.— Москва: Издательство Юрайт, 2024.— 192с.

https://urait.ru/quiz/run-test/6B4E520E-F063-47C3-A0CD-97F991B9D0B3/CC16C103-DDAF-4BC8-8648-01654ED59129

Тема 7. Разработка пользовательского интерфейса

Гниденко, И.Г. Технология разработки программного обеспечения: учебное пособие для среднего профессионального образования/ И.Г.Гниденко, Ф.Ф.Павлов, Д.Ю.Федоров.— 2-е изд., перераб. и доп.— Москва: Издательство Юрайт, 2023.— 248с

https://urait.ru/quiz/run-test/C852CBFD-96EA-413E-9025-4F49ABCB4FFA/347ADD07-F2DF-4F4F-957F-187EC8DA73DD

МДК.01.02. Поддержка и тестирование программных модулей

Тема 1. Отладка и тестирование программного обеспечения

Лаврищева, Е.М.

Программная инженерия и технологии программирования сложных систем: учебник для вузов/ Е.М.Лаврищева.— 2-е изд., испр. и доп.— Москва: Издательство Юрайт, 2023.— 432с.

https://urait.ru/quiz/run-test/0C1033FA-321D-40BA-A205-9492B0831AFE/9652EEB2-FB26-4DD8-9279-CB1979E1398A

Тема 2. Документирование

Колошкина, И.Е.

Автоматизация проектирования технологической документации: учебник и практикум для среднего профессионального образования/ И.Е.Колошкина.— Москва: Издательство Юрайт, 2023.— 371с.— (Профессиональное образование).— ISBN978-5-534-13635-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 30 https://urait.ru/quiz/run-test/1CD114D2-FAA8-47C3-8C05-A081DA2B3A2C/BFCC0532-ABCE-4CE7-97D1-B96DD7839DF1

МДК 01.03. Разработка мобильных приложений

Тема 1. Основные платформы и языки разработки мобильных приложений.

Соколова, В.В.

Разработка мобильных приложений: учебное пособие для среднего профессионального образования/ В.В.Соколова.— Москва: Издательство Юрайт, 2023.— 160с.— (Профессиональное образование).— ISBN978-5-534-16868-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 39

https://urait.ru/quiz/run-test/1E71DB51-F442-47C2-8D0B-82988CF11339/51B6B6DD-C48A-4EC0-A55E-D53A7A85FBEE

Тема 2. Создание и тестирование модулей для мобильных приложений

Соколова, В.В.

Разработка мобильных приложений: учебное пособие для среднего профессионального образования/ В.В.Соколова.— Москва: Издательство Юрайт, 2023.— 160с.— (Профессиональное образование).— ISBN978-5-534-16868-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 39

https://urait.ru/quiz/run-test/C19F1E16-9387-4683-8964-D03B3DF2C146/E9446C25-564A-4AC4-BDCB-B2435514E98D

МДК 01.04. Системное программирование

Тема 1. Программирование на языке низкого уровня

Новожилов, О.П.

Архитектура ЭВМ и систем: учебник для вузов/ О.П.Новожилов.— 2-е изд., испр. и доп.— Москва: Издательство Юрайт, 2024.— 511с.— (Высшее образование).— ISBN978-5-534-18445-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 122

https://urait.ru/quiz/run-final-test/4F89AC1B-7BEE-43BA-9EF0-8A307912B796

3.2 Критерии оценки результатов выполнения тестовых заданий

В качестве критерия освоения материала профессионального модуля берётся выполнение 50% заданий от их общего числа в тест-задании.

Для интерпретации результатов тестирования используется переводная шкала, с помощью которой результаты теста преобразуют в оценку знаний студентов по пятибалльной системе. Переводная шкала представлена в Таблице 1.

Таблица 1 – Переводная шкала для интерпретации результатов выполнения тест-задания

% ошибочных ответов	**Интерпретация результатов тестирования по 5-ой системе**
до 10%	5
от 15% до 30%	4
от 35% до 50%	3
более 50%	2

3.3 Письменная форма контроля

МДК 01.01. Разработка программных модулей

ТЕМА	**НАЗВАНИЕ ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ РАБОТ**	**САМОСТОЯТЕЛЬНАЯ РАБОТА**
1	-	доклад «Модели жизненного цикла программного обеспечения»
2	Оценка сложности алгоритмов сортировки. Оценка сложности алгоритмов поиска. Оценка сложности рекурсивных алгоритмов. Оценка сложности эвристических алгоритмов.	доклад «Неразрешимые алгоритмические задачи»
3	Работа с классами. Перегрузка методов. Определение операций в классе. Создание наследованных классов. Работа с объектами через интерфейсы. Использование стандартных интерфейсов. Работа с типом данных структура. Коллекции. Параметризованные классы. Использование регулярных выражений.	Создание диаграммы классов

	Операции со списками.	
4	Использование основных шаблонов. Использование порождающих шаблонов. Использование структурных шаблонов. Использование поведенческих шаблонов.	-
5	Разработка приложения с использованием текстовых компонентов Разработка приложения с несколькими формами. Разработка приложения с не визуальными компонентами. Разработка игрового приложения. Разработка приложения с анимацией.	Создание глоссария по теме
6	Оптимизация и рефаторинг кода	-
7	Разработка интерфейса пользователя	эскизы пользовательского интерфейса
8	Создание приложения с БД. Создание запросов к БД. Создание запросов к БД. Создание хранимых процедур.	создание интеллект-карты по теме

МДК 01.02. Поддержка и тестирование программных модулей

ТЕМА	**НАЗВАНИЕ ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ РАБОТ**	**САМОСТОЯТЕЛЬНАЯ РАБОТА**
1	Тестирование «белым ящиком». Тестирование «черным ящиком». Модульное тестирование. Интеграционное тестирование	отчет по тестированию программы
2	Оформление документации на программные средства с использованием инструментальных средств	отчет по тестированию программы

МДК 01.03 Разработка мобильных приложений

ТЕМА	**НАЗВАНИЕ ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ РАБОТ**	**САМОСТОЯТЕЛЬНАЯ РАБОТА**
1	Установка инструментария для разработки мобильных приложений. Настройка среды для разработки мобильных приложений. Установка среды разработки мобильных приложений с применением виртуальной машины.	доклад «Инструменты разработки мобильных приложений»
2	Создание эмуляторов и подключение устройств. Настройка режима терминала. Создание нового проекта. Изучение и комментирование кода. ЛР «Изменение элементов дизайна» Обработка событий: подсказки.	доклад «Способы тестирования мобильных приложений»

	Обработка событий: цветовая индикация. Подготовка стандартных модулей. Обработка событий: переключение между экранами. Передача данных между модулями. Тестирование и оптимизация мобильного приложения	

МДК 01.04 Системное программирование

ТЕМА	НАЗВАНИЕ ПРАКТИЧЕСКИХ И ЛАБОРАТОРНЫХ РАБОТ	САМОСТОЯТЕЛЬНАЯ РАБОТА
1	Использование потоков. Обмен данными. Сетевое программирование сокетов. Работы с буфером экрана.	создание глоссария по теме

3.4. Критерии оценки результатов выполнения письменных заданий

Оценка «5» (отлично) – краткая, ясная и четкая описание решения задания, верно, на высоком профессиональном уровне, с правильными пояснениями. Обоснованность ответа. Необходимо мотивировать выбранный курс действий, приводящих к разрешению ситуации, и объяснить причины и рациональность его выбора.

Оценка «4» (хорошо) – задание выполнено в целом верно, но с небольшими неточностями, имеются неточности в пояснении.

Оценка «3» (удовлетворительно) – задание выполнено верно, но не доведен до конца, либо в нем имеются ошибки, которые, однако, не приводят к принципиально неверному решению.

Оценка «2» (неудовлетворительно) – задание не выполнено или выполнено неверно

3.5 Устная форма контроля

МДК 01.01. Разработка программных модулей

Примерный список вопросов для устного ответа по темам 1-8.

1. В чем сущность объектной декомпозиции?

2. Для чего используют язык UML? Почему его называют языком моделирования? Чем обусловлен выбор именно этого языка в качестве стандарта описания объектных разработок?

3. Какие диаграммы используют в качестве спецификаций программного обеспечения при объектном подходе?

4. Что такое «вариант использования»? Как строится диаграмма вариантов использования, и какую информацию она содержит?

5. Как описывают структуру программного обеспечения при объектном подходе? Что такое «пакет»? Для чего используют диаграммы пакетов?

6. Что понимают под диаграммой классов? Какую информацию она содержит?

7. Какие отношения между классами рассматриваются в диаграмме классов?

8. Что понимают под диаграммой компонентов? Какую информацию она содержит? В каких случаях целесообразно строить диаграммы компонентов?

9. Какую информацию содержит диаграмма размещения? В каких случаях целесообразно использовать эти диаграммы?

МДК 01.02. Поддержка и тестирование программных модулей

Примерный список вопросов для устного ответа по темам 1-2.

1 Введение: тестирование - способ обеспечения качества программного продукта

2. Основные понятия тестирования

3. Подходы к обоснованию истинности формул и программ и их связь с тестированием. Вопросы организации тестирования. Фазы тестирования, основные проблемы тестирования и поставлена задача выбора конечного набора тестов.

4. Требования к идеальному критерию тестирования и классы частных критериев. Особенности применения структурных и функциональных

критериев на базе конкретных примеров. Особенности применения методов стохастического тестирования и метод оценки скорости выявления ошибок. Мутационный критерий и на примере иллюстрируется техника работы с ним.

5. Оценка оттестированности проекта: метрики и методика интегральной оценки

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

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

8. Интеграционное тестирование и его особенности для объектноориентированного программирования

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

10. Разновидности тестирования: системное и регрессионное тестирование

11. Автоматизация тестирования

12. Особенности индустриального тестирования

13. Документирование и оценка индустриального тестирования

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

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

МДК 01.03 Разработка мобильных приложений

Примерный список вопросов для устного ответа по темам 1-2.

1 Архитектура платформы Android. Уровень ядра. Уровень библиотек.

2. Архитектура платформы Android. Dalvik Virtual Machine.

3. Архитектура платформы Android. Уровень каркаса приложений. Уровень приложений.

4. Среда разработки для Android. Eclipse IDE. Плагин ADT. Android Virtual Device.

5. Android SDK. Версии SDK и Android API Level.

6. Структура проекта Android-приложения в Eclipse. Каталоги ресурсов. Файл R.java.

7. Графический интерфейс пользователя в Android-приложениях. XMLpa3MeiKa интерфейса.

8. Архитектура платформы Android.

9. XMI .-разметка интерфейса пользователя.

10. XAMI .-разметка интерфейса пользователя.

11. Базовые элементы управления.

12. Ресурсы в Android-приложениях.

13. Ресурсы в Windows Phone-приложениях.

14. Активности и интенты.

15. Обработка пользовательского ввода. Касания, ввод текста.

16. Типы компоновок графического интерфейса. FraneLayout, LinearLayout, TableLayout, RelativeLayout.

17. Базовые элементы управления. TextView. EditText. Тип ввода текста. Параметры отображения клавиатуры. ImageView.

18. Диалоговые окна. AlertDialog. ProgressDialog. DatePickerDialog. TimePickerDialog. Создание пользовательских диалоговых окон.

19. Многопоточные приложения в Android и Windows Phone. Использование системных таймеров и системного времени.

20. Процессы в Android. Объекты Activity. Состояния Activity.

21. Использование объектов Intent. Intent-фильтры.

22. Использование ресурсов. Ссылки на ресурсы. Загрузка простых типов из ресурсов. Загрузка файлов произвольного типа.

23. Файловая система Android. Чтение и запись файлов.

24. Адаптеры данных. Отображение данных в компонентах ListView, GridView, AutoCompleteTextView, MultiAutoCompleteTextView.

25. Пользовательские настройки. Использование SharedPreferences. Виды настроек.

26. Работа с графикой. Drawable и Canvas.

27. Работа с анимацией. Tween Animation и Frame Animation. Описание анимации в XML и в коде программы.

28. Службы в Android. Компонент Service.

29. Датчики мобильных устройств. Управление датчиками в приложении. Виды датчиков и особенности их использования.

30. Программный доступ к дисплею устройства. Менеджер окон. Параметры дисплея.

МДК 01.04 Системное программирование

Примерный список вопросов для устного ответа по теме1.

1 Объясните, как Вы понимаете термин системное программное обеспечение сформулируете основные понятия.

2. Расскажите, как реализуются арифметические операции на Ассемблере.

3. Объясните, как Вы понимаете команды обработки строковых данных.

4. Перечислите и охарактеризуйте средства аппаратной поддержки функций ОС.

5. Объясните двоичное кодирование информации. Представление элементарных типов данных: натуральные числа, целые числа со знаком, числа с плавающей точкой.

6. Расскажите про компилятор ассемблер - программ, редактор связей (загрузчик).

7. Объясните общую структуру машинных команд.

8. Как организуется ввод - вывод и классифицируются внешние устройства.

9. Назовите и опишите константы, метки и условную компиляцию.

10. Перечислите и опишите этапы проектирования и выполнения программ

11. Опишите задачи, для решения которых используется язык ассемблера.

12. Объясните регистры eax, ebx, ecx, edx и их специальные свойства.

13. Расскажите про команды ADC и SBB.

14. Дайте определения резидентным программам.

15. Расскажите про команды MUL и IDIV.

3.6 Критерии оценки результатов выполнения устных заданий

Оценка «5» (отлично) предполагает грамотное и логичное изложение ответа в устной форме, на практикоориентированные вопросы студент дает обоснование собственного ответа с точки зрения известных теоретических положений.

Оценка «4» (хорошо) – если студент полно освоил учебный материал, владеет научнопонятийным аппаратом, ориентируется в изученном материале, осознанно применяет теоретические знания на практике, грамотно излагает ответ, но содержание и форма ответа имеют отдельные неточности.

Оценка «3» (удовлетворительно) – если студент обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, в применении теоретических знаний, при ответе на практико-ориентированные вопросы;

Оценка «2» (неудовлетворительно) – если студент имеет разрозненные, бессистемные знания, допускает ошибки в определении базовых понятий, искажает их смысл; не может практически применять теоретические знания.

.

4. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Промежуточная аттестация по профессиональному модулю ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем» состоит из следующих компонентов:

семестровый зачет - МДК 01.01 (IV семестр), МДК 01.02 (V семестр);

дифференцированный зачет – МДК 01.01 (V семестр);

комплексный экзамен - МДК 01.02 +МДК 01.03 (VI семестр)

экзамен – МДК 01.03 (VI семестр), МДК 01.04 (VII семестр)

Вопросы к семестровому зачету по МДК 01.01:

1. Этапы решения задач на компьютере.

2. Понятие алгоритма. Подходы к определению алгоритма. Свойства алгоритма. Способы записи алгоритма.

3. Понятие алгоритма. Понятие исполнителя. Система команд исполнителя.

4. Языки программирования. Алгоритмические языки (алфавит, синтаксис, семантика). Способы описания синтаксиса (язык металингвистических формул, синтаксические диаграммы).

5. Интегрированная среда разработки приложений.

6. Структура программы, элементы языка (алфавит). Понятие типа данных.

7. Операции (арифметические, логические) на типах. Стандартные функции. Выражения.

8. Процедуры консольного ввода и вывода, управление вводом-выводом. Оператор присваивания. Совместимость по присваиванию.

9. Условный оператор. Оператор множественного ветвления (выбора).

10. Циклы: с предусловием, с постусловием. Связь с другими циклами.

11. Циклы: с параметром. Связь с другими циклами.

12. Структурированные типы данных.

13. Сортировка массивов.

14. Подпрограммы.

15. Процедуры и функции.

16. Простые типы данных.

17. Множества.

18. Комбинированный тип данных (записи).

19. Файловые типы.

20. Текстовые файлы.

21. Прямой и последовательный доступ к компонентам файла.

22. Поиск в типизированных файлах

23. Типизированные файлы.

24. Статическая и динамически распределяемая память.

25. Динамические структуры данных.

26. Модуль. Общая структура модуля. Компиляция и подключение модуля.

27. Объектно-ориентированное программирование. Компоненты.

Как получить список всех атрибутов объекта?

28. Как получить список всех публичных атрибутов объекта?

29. Как получить список методов объекта?

30. В какой "магической" переменной хранится содержимое help?

31. Есть два кортежа, получить третий как конкатенацию первых двух

32. Есть два кортежа, получить третий как объединение уникальных элементов первых двух кортежей

33. Почему если в цикле меняется список, то используется for x in lst[:], что означает [:]?

34. Есть два списка одинаковой длины, в одном ключи, в другом значения. Составить словарь.

35. Есть два списка разной длины, в одном ключи, в другом значения. Составить словарь. Для ключей, для которых нет значений использовать None в качестве значения. Значения, для которых нет ключей игнорировать.

36. Есть словарь. Инвертировать его. Т.е. пары ключ: значение поменять местами — значение: ключ.

37. Есть строка в юникоде, получить 8-битную строку в кодировке utf-8 и cp1251

38. Есть строка в кодировке cp1251, получить юникодную строку

Вопросы к семестровому зачету по МДК 01.02:

1 Введение: тестирование - способ обеспечения качества программного продукта

2. Основные понятия тестирования

3. Подходы к обоснованию истинности формул и программ и их связь с тестированием. Вопросы организации тестирования. Фазы тестирования, основные проблемы тестирования и поставлена задача выбора конечного набора тестов.

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

5. Оценка оттестированности проекта: метрики и методика интегральной оценки

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

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

8. Интеграционное тестирование и его особенности для объектноориентированного программирования

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

10. Разновидности тестирования: системное и регрессионное тестирование

11. Автоматизация тестирования

12. Особенности индустриального тестирования

13. Документирование и оценка индустриального тестирования

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

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

16. Регрессионное тестирование: разновидности метода отбора тестов

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

18. Регрессионное тестирование: алгоритм и программная система поддержки

19. Описание тестируемой системы и ее окружения. Планирование тестирования

20. Модульное тестирование на примере классов

21. Интеграционное тестирование

22. Системное тестирование

23. Ручное тестирование

24. Автоматизация тестирования с помощью скриптов

25. Автоматическая генерация тестов на основе формального описания

26. Описание ручного тестирования

27. Автоматизация тестирования с помощью скриптов

28. Описание автоматической генерации MSC тестов

4.1. Критерии оценивания знаний, обучающихся на семестровом зачете

До даты проведения с/зачета необходимо пройти оценивание всех работ, выполненных в течение VI семестра: теоретическое обучение, практические занятия, задания для самостоятельного выполнения.

К установленной дате с/зачета обучающийся должен выполнить 100% работ, предусмотренных рабочей программой профессионального модуля ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем».

Оценки с/зачета – зачтено, незачет.

Оценка «зачтено» выставляется студенту, проявившему знания основного программного материала в объеме, необходимом для последующего обучения и предстоящей практической деятельности,

знакомому с основной рекомендованной литературой, но допустившему неполные или слабо аргументированные ответы, испытывающему затруднения.

Оценка «не зачтено» выставляется студенту, обнаружившему значительные пробелы в знании программного материала, допустившему принципиальные ошибки при применении теоретических знаний, которые не позволяют ему продолжить обучение или приступить к практической деятельности без дополнительной подготовки по данной дисциплине.

4.2 Критерии оценивания знаний, обучающихся на дифференцированном зачете

До даты проведения дифференцированного/зачета необходимо пройти оценивание всех работ, выполненных в течение V семестра: теоретическое обучение, практические занятия, задания для самостоятельного выполнения.

К установленной дате д/зачета обучающийся должен выполнить 100% работ, предусмотренных рабочей программой профессионального модуля ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем».

Дифференцированный зачет оценивается по пятибалльной шкале путем вычисления среднего балла успеваемости.

4.3 Критерии оценивания знаний, обучающихся на экзамене

Вопросы к экзамену по МДК.01.01

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. SQLзапросы на выборку, добавление, редактирование и удаление данных. Агрегатные функции. Группировка записей.

35. Статические и динамические параметрические SQL-запросы

Практические задания к экзамену по МДК 01.01

1. Подсчитать сумму элементов над главной диагональю квадратной матрицы и под ней.

2. Определить максимальный и минимальный элементы двумерного массива и их индексы.

3. Заполнить одномерный массив случайными неповторяющимися числами.

4. Написать программу, реализующую игру «Угадай число».

5. Написать программу решения квадратного уравнения.

6. Расположить строки двумерного массива в порядке возрастания сумм элементов в них.

7. Заполнить одномерный массив случайными числами из интервала [-10, 10] . Упорядочить массив по возрастанию, если сумма элементов массива положительна, или по убыванию в противном случае.

8. Сохранить данные одномерного массива в файле данных. Организовать чтение данных из файла.

9. Решить линейное неравенство, не используя условную конструкцию.

10. Определить количество слов во введенном тексте.

11. Заполнить двумерный массив случайными числами. Вывести массив на экран. Найти максимальный и минимальный элементы, их индексы.

12. Заполнить одномерный массив случайными числами. Упорядочить массив по возрастанию или убыванию по желанию пользователя.

13. Подсчитать, сколько среди пятизначных чисел таких, у которых сумма четных цифр равна сумме нечетных цифр.

14. Определить, сколько раз в тексте встречается указанный фрагмент.

15. Создать проект, позволяющий сохранять произвольный текст в текстовом файле с указанным пользователем именем. Загружать текст из файла.

16. Создать проект, позволяющий сохранять текст, набранный в текстовом окне в файле данных, очищать текстовое окно и загружать в него сохраненный текст.

17. Создать проект, позволяющий построить треугольник по

координатам вершин.

18. Имеется десять флажков. Создать проект, позволяющий сохранять состояние флажков, обнулять и восстанавливать сохраненное состояние.

19. Организовать анимацию процесса движения шарика по окружности.

20. Создать проект, реализующий часы с «будильником».

21. Проект содержит три текстовых окна. Используя буфер обмена, организовать обмен информацией между приложениями.

22. Вывести алфавит введенного пользователем текста.

23. Создать проект, позволяющий добавлять записи в простой список и удалять их.

24. Расположить слова строкового массива в порядке возрастания их длин.

25. Построить на экране правильный N-угольник. Количество вершин и радиус окружности, на которой они расположены, указываются пользователем.

26. Построить график функции в выбранной пользователем системе координат.

27. Составить программу, генерирующую цвет по трем его составляющим (красный, зеленый, синий).

28. Составить программу, моделирующую работу светофора.

29. Разработать класс, объекты которого обладают методами, позволяющими вывести на экран в любую его точку правильный многоугольник произвольного размера с произвольным количеством вершин.

30. Написать программу, моделирующую движение броуновской частицы.

31. Разработать класс, объекты которого обладают методами, позволяющими вывести на экран график функции. График должен выводиться в прямоугольную область указанного размера. Объект должен иметь свойство, определяющее величину единичного отрезка.

32. Разработать класс, объекты которого фиксируют время своего

существования.

33. Разработать класс, объекты которого обладают методами, позволяющими вывести на экран анимацию вращающегося шарика. Координата точки – оси вращения, скорость вращения, радиус вращения и направление вращения задаются.

34. Описать процедуру, имеющую три аргумента целого типа и работающую по следующему принципу: значения трех переменных целого типа, описанных выше точки вызова и указанных в качестве аргументов процедуры, должны быть расположены по возрастанию в порядке указания аргументов в процедуре.

35. Разработать класс, объекты которого обладают методами, позволяющими заполнить одномерный массив случайными неповторяющимися числами, вывести его на экран и упорядочить.

Вопросы к экзамену по МДК.01.02

1. Перечислите основные характеристика качества программного обеспечения.

2 Расскажите о стандартах качества программного обеспечения.

3 Расскажите о методах контроля качества программного обеспечения. Поясните разницу между валидацией и верификацией.

4 Дайте определение понятий «ошибка», «дефект», «сбой». Поясните разницу между ними.

5 Дайте определение понятий «отладка» и «тестирование», поясните разницу между ними.

6 Перечислите виды тестирования, приведите примеры.

7 Расскажите об уровнях тестирования.

8 Расскажите об основных принципах тестирования и о критериях завершения тестирования.

9 Расскажите о стратегиях тестирования, поясните разницу между ними.

10.Расскажите о методах тестирования, относящихся к стратегии белого ящика. Поясните разницу между ними.

11.Расскажите о методах тестирования, относящихся к стратегии черного ящика. Поясните разницу между ними.

12.Расскажите о методе покрытия операторов, приведите примеры.

13.Расскажите о методе покрытия условий, приведите примеры.

14.Расскажите о методе комбинаторного покрытия условий, приведите примеры.

15.Расскажите о методе граничных значений, приведите примеры.

16.Расскажите о методе разбиения на классы эквивалентности, приведите примеры.

17.Расскажите о необходимости документирования тестирования. Перечислите, какие сведения необходимо указывать в тест-кейсах.

18.Расскажите о системах отслеживания ошибок. Назовите известные вам системы.

19.Расскажите о возможности разработки модульных тестов (unit-тестов) в среде программирования.

20.Расскажите о возможностях класса Assert для автоматизации тестирования.

21.Расскажите о технологии «разработки через тестирование», приведите примеры.

22.Дайте определение термина «надежность программного обеспечения», расскажите о возможностях оценки надежности.

Практические задания к экзамену МДК 01.02

Провести тестирование программы, решающей задачу по варианту:

1. Подсчитать сумму элементов над главной диагональю квадратной матрицы и под ней.

2. Определить максимальный и минимальный элементы двумерного массива и их индексы.

3. Заполнить одномерный массив случайными неповторяющимися

числами.

4. Написать программу, реализующую игру «Угадай число».

5. Написать программу решения квадратного уравнения.

6. Расположить строки двумерного массива в порядке возрастания сумм элементов в них.

7. Заполнить одномерный массив случайными числами из интервала [-10, 10] . Упорядочить массив по возрастанию, если сумма элементов массива положительна, или по убыванию в противном случае.

8. Сохранить данные одномерного массива в файле данных. Организовать чтение данных из файла.

9. Решить линейное неравенство, не используя условную конструкцию.

10. Определить количество слов во введенном тексте.

11. Заполнить двумерный массив случайными числами. Вывести массив на экран. Найти максимальный и минимальный элементы, их индексы.

12. Заполнить одномерный массив случайными числами. Упорядочить массив по возрастанию или убыванию по желанию пользователя.

13. Подсчитать, сколько среди пятизначных чисел таких, у которых сумма четных цифр равна сумме нечетных цифр.

14. Определить, сколько раз в тексте встречается указанный фрагмент.

15. Создать проект, позволяющий сохранять произвольный текст в текстовом файле с указанным пользователем именем. Загружать текст из файла.

16. Создать проект, позволяющий сохранять текст, набранный в текстовом окне в файле данных, очищать текстовое окно и загружать в него сохраненный текст.

17. Создать проект, позволяющий построить треугольник по координатам вершин.

18. Имеется десять флажков. Создать проект, позволяющий сохранять состояние флажков, обнулять и восстанавливать сохраненное состояние.

19. Организовать анимацию процесса движения шарика по окружности.

20. Создать проект, реализующий часы с «будильником».

21. Проект содержит три текстовых окна. Используя буфер обмена, организовать обмен информацией между приложениями.

22. Вывести алфавит введенного пользователем текста.

23. Создать проект, позволяющий добавлять записи в простой список и удалять их.

24. Расположить слова строкового массива в порядке возрастания их длин.

25. Построить на экране правильный N-угольник. Количество вершин и радиус окружности, на которой они расположены, указываются пользователем.

26. Построить график функции в выбранной пользователем системе координат.

27. Составить программу, генерирующую цвет по трем его составляющим (красный, зеленый, синий).

28. Составить программу, моделирующую работу светофора.

29. Разработать класс, объекты которого обладают методами, позволяющими вывести на экран в любую его точку правильный многоугольник произвольного размера с произвольным количеством вершин.

30. Написать программу, моделирующую движение броуновской частицы.

31. Разработать класс, объекты которого обладают методами, позволяющими вывести на экран график функции. График должен выводиться в прямоугольную область указанного размера. Объект должен иметь свойство, определяющее величину единичного отрезка.

32. Разработать класс, объекты которого фиксируют время своего существования.

33. Разработать класс, объекты которого обладают методами, позволяющими вывести на экран анимацию вращающегося шарика.

Координата точки – оси вращения, скорость вращения, радиус вращения и направление вращения задаются.

34. Описать процедуру, имеющую три аргумента целого типа и работающую по следующему принципу: значения трех переменных целого типа, описанных выше точки вызова и указанных в качестве аргументов процедуры, должны быть расположены по возрастанию в порядке указания аргументов в процедуре.

35. Разработать класс, объекты которого обладают методами, позволяющими заполнить одномерный массив случайными неповторяющимися числами, вывести его на экран и упорядочить.

36. Решить квадратное уравнение, используя директивы препроцессора и условную операцию.

37. Описать процедуру, позволяющую произвести обмен значениями переменных указанных в качестве аргумента.

38. Разработать компонент, предлагающий интерфейс для отображения одномерного массива и имеющий методы для заполнения, очистки и сортировки.

39. Разработать компонент «Переключатель», имеющий пять стационарных состояний. Компонент должен иметь свойство для установки состояния и обрабатывать событие, связанное с выбором другого состояния.

40. На форме имеется десять командных кнопок (пронумерованных от 1 до 10) и графический контейнер. При нажатии на одну из кнопок в контейнере должно появиться соответствующее римское число.

41. Написать программу, позволяющую просматривать записи простейшей базы данных, добавлять, корректировать и удалять их.

42. Разработать программу в Lazarus, позволяющую заполнять двумерный массив с указанным пользователем количеством срок и столбцов и находить среднее арифметическое в каждой строке и в каждом столбце.

43. Используя общий диалог, написать программу, позволяющую сохранять указанный пользователем текст в файле и считывать сохраненный

текст из файла.

44. Используя объектно-ориентированный подход, написать программу, моделирующую движение броуновских частиц.

45. Разработать компонент «Таймер», выводящий время, прошедшее с момента запуска таймера. Компонент должен иметь метод, возвращающий текущее время, обрабатывать событие, связанное с включением и выключением таймера и метод, обнуляющий таймер.

46. Описать класс, объекты которого связываются с текстовыми окнами, хранят введенную пользователем в окно информацию, обладают методом сортировки символов в текстовых окнах в алфавитном порядке и методом восстанавливающем текст в окне после сортировки.

47. Разработать компонент, предлагающий интерфейс для ввода данных и вывода решения квадратного уравнения.

48. Создать проект, моделирующий работу секундомера. Секундомер должен иметь циферблат с минутной и секундной стрелками, метку и одну командную кнопку.

49. Необходимо написать консольную программу, где пользователь будет вводить с клавиатуры 2 числа. Числа будут сравниваться с последующим выводом в консоль результата этого сравнения (равны ли значения, а если нет, то какое число больше/меньше).

50. Пусть пользователь имеет возможность ввести одно число, причем: когда значение больше пяти и одновременно с этим меньше десяти, выводится один текст с соответствующим сообщением; когда значение не попадает в указанный диапазон, программа его не узнает, выводя другой текст с соответствующим сообщением.

Вопросы к экзамену по МДК.01.03

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-манифсст

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

Практические задания к экзамену МДК 01.03

1 Разработать приложение, позволяющее проигрывать аудиофайлы на устройстве.

2 Разработать приложение «Записная книжка» (создание, удаление, поиск заметок, сохранение в базе SQLite / либо файловая система).

3 Разработать приложение «Калькулятор».

4 Разработать программу, сохраняющую изображение с камеры.

5 Разработать приложение «Школьный дневник».

6 Реализовать приложение, отображающее новости и состоящее из следующих экранов: 1 Экран списка новостей. Пользователь может отфильтровать новости по темам (политика, спорт, наука, технологии и т.д.) 2 Экран просмотра новости. 3 Экран просмотра прикрепленных к новости изображений. Новости должны предоставляться сторонним сервисом через Интернет. В приложении должна быть возможность обновить новости по команде пользователя. Между запусками приложения тексты загруженных новостей должны сохраняться. Приложение должно корректно работать при отсутствии сети и показывать ранее загруженные новости.

7 Разработайте мобильное приложение по работе с анимацией в Android. Предусмотрите возможность выбора цвета интерфейса или картинки.

Вопросы к экзамену по МДК.01.04

1. Глобальные и локальные идентификаторы. Видимость объектов (идентификаторов). Примеры.

2. Дайте характеристику языку ассемблера, определение, преимущества.

3. Двумерные массивы (списки списков). Вложенная генерация.

4. Для чего используются трансляторы, компиляторы, интерпретаторы.

5. Интегрированная среда разработки приложений.

6. Классы в Python. Конструктор класса.

7. Классы в Python. Наследование.

8. Классы в Python. Перегрузка операторов.

9. Комбинированный тип данных (записи). Оператор присоединения. Записи с вариантами. C#

10. Кортежи в Python. Создание и использование. Кортежи переменных

11. Литералы чисел и строк в Python

12. Модуль. Общая структура модуля. Компиляция и подключение модуля. C#

13. Назовите и опишите константы, метки и условную компиляцию.

14. Область действия переменных и других идентификаторов.

15. Объектно-ориентированное программирование. Компоненты.

16. Объясните двоичное кодирование информации. Представление элементарных типов данных: натуральные числа, целые числа со знаком, числа с плавающей точкой.

17. Негистры eax, ebx, ecx, edx и их специальные свойства.

18. Операции (арифметические, логические) на типах. Стандартные функции. Выражения. C#

19. Опишите назначение простейших команд Ассемблера. Расскажите про команды ADC и SBB.

20. Параметры-переменные, параметры-значения, параметры-константы. Примеры.

21. Перечислите и опишите этапы проектирования и выполнения

программ

22. Перечислите и охарактеризуйте средства аппаратной поддержки функций ОС.

23. Подпрограммы. Основные способы передачи параметров в подпрограмму, их сравнение. C#

24. Понятие алгоритма. Подходы к определению алгоритма. Свойства алгоритма. Способы записи алгоритма. Понятие исполнителя. Система команд исполнителя.

25. Понятие о структурном программировании. Другие парадигмы программирования: сравнительная характеристика.

26. Понятие процесса и потока: раскройте и охарактеризуйте.

27. Принципы структурного программирования. Блочный (модульный) подход к построению алгоритмов решения задач.

28. Простые типы данных. Структурированные типы данных. Строковый тип данных, примеры. C#

29. Процедуры, ориентированные на работу со строками. Примеры.

30. Процедуры. Вызов процедуры в программах. Формальные и фактические параметры. Соответствие между параметрами. Примеры.

31. Процедуры. Организация и вызов. Примеры. C#

32. Расскажите как формируются предложения на Ассемблере из лексем. Опишите операнды Ассемблера

33. Расскажите про команды MUL, IDIV, IMUL и DIV.

34. Расскажите про компилятор ассемблер - программ, редактор связей (загрузчик).

35. Расскажите про обработку прерываний. Контекст процесса (потока). Особенности операционных систем реального времени.

36. Рекурсия. Механизм рекурсии: действия на рекурсивном спуске и рекурсивном возврате. Понятие о глубине рекурсии. Примеры.

37. Символьный тип данных. Строковый тип данных. Строковые переменные и константы. Двойственный характер строк.

38. Сортировка массивов. C#

39. Списки в Python. List comprehensions: генерация списков.

40. Списки в Python. Методы списков и операции со списками

41. Срезы списков. Присваивание в срез. Проблема копирования списка

42. Строки в Python. Методы find, count, replace, strip, upper, lower и другие.

43. Строки в Python. Срезы с двумя и тремя параметрами

44. Структура программы, элементы языка (алфавит). Понятие типа данных. C#

45. Структурированные типы данных. Линейные массивы. Примеры задач. Двумерные массивы. Примеры задач. C#

46. Условный оператор и каскадная условная конструкция в Python

47. Условный оператор. Оператор множественного ветвления (выбора). C#

48. Файловые типы. Общие процедуры для работы с файлами. Компонентные (типизированные) файлы. C#

49. Функции, ориентированные на работу со строками. Примеры.

50. Функции. Их отличие от процедур. Способ обращения к функции. Примеры.

51. Функции. Организация и вызов. Примеры. C#

52. Цикл for и его особенности в Python. Функция range()

53. Циклы: с параметром. Связь с другими циклами. C#

54. Циклы: с предусловием, с постусловием. Связь с др. циклами. C#

55. Языки программирования. Алгоритмические языки (алфавит, синтаксис, семантика). Способы описания синтаксиса (язык металингвистических формул, синтаксические диаграммы).

Практические задания к экзамену МДК 01.04

1. Разработать программу, которая перезапускает себя по двойному щелчку мыши. Одновременное нахождение в памяти двух экземпляров этой программы недопустимо.

2. Разработать DLL-библиотеку, экспортирующую функции выделения

и освобождения памяти DllAlloc и DllFree. Разработать тестовую программу с динамическим импортом этой DLL.

3. Разработать программу, которая рисует в окне с помощью мыши. Если кнопка нажата, выполняется рисование.

4. Разработать программу, рисующую на экране прямоугольник с четырьмя случайными цифрами PIN-кода. (Защита, применяемая при регистрации электронных адресов на Web-сайтах.)

5. Разработать программу "часы", которая выводит в окне время.

6. Разработать программу, которая по таймеру с периодичностью 1 сек. проверяет, запущен ли процесс "spy", и если запущен, убивает его.

7. Разработать программу, не допускающую запуск своего второго экземпляра.

8. Написать минимальную GUI-программу, рисующую эллипс, вписанный в рабочую область окна.

9. Вторичному потоку требуется вызвать процедуру Update в контексте главного UI-потока (например, для обновления списка на экране). Решить поставленную задачу.

10. Разработать программу, которая проецирует файл в память и ищет в нем запись с полем Number равным 10. Структура записи: Number: Integer, Offset: Pointer. Number – данные элемента списка. Offset – смещение в файле до следующего элемента списка. Элементы списка в файле могут идти в произвольном порядке. Реальный логический порядок определяется полем Offset. У последнего элемента списка поле Offset равно 0.

11. Разработать контейнер Array, который можно использовать в многопоточной среде.

12. Разработать контейнер List, который можно использовать в многопоточной среде.

13. Разработать программу, выводящую в окне растровое изображение из некоторого файла.

14. Разработать DLL-библиотеку, экспортирующую функции работы со

строками StrLength и StrCopy. Разработать 2 тестовые программы: одна подключает DLL методом статического импорта, вторая – методом динамического импорта.

15. Разработать программу, рисующую на экране картинку из файла. С помощью клавиш управления курсором картинка должна перемещаться в окне.

16. Разработать программу, которая по двойному щелчку мыши позволяет перетаскивать свое окно на экране с помощью клавиш управления курсором.

17. Разработать DLL-библиотеку, экспортирующую класс-контейнер Array.

18. Разработать программу, которая по двойному щелчку запускает свою вторую копию.

Ответы на теоретические вопросы оцениваются по пятибалльной шкале следующим образом:

«Отлично» - теоретическое содержание курса освоено полностью, без пробелов, сформированы, все практические задания выполнены, качество их выполнения оценено высоко.

«Хорошо» - теоретическое содержание курса освоено полностью, без пробелов, некоторые умения сформированы недостаточно, все практические задания выполнены, некоторые виды заданий выполнены с ошибками.

«Удовлетворительно» - теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые умения работы с освоенным материалом в основном сформированы, большинство заданий выполнено, некоторые из выполненных заданий содержат ошибки.

«Неудовлетворительно» - теоретическое содержание курса не освоено, необходимые умения не сформированы, практические задания содержат грубые ошибки

Выполнение практического задания оценивается по пятибалльной шкале следующим образом:

Оценка «5» (отлично) - краткая, ясная и четкая констатация факта или события в ситуации, выход из ситуации найден, верно, на высоком профессиональном уровне, с правильными пояснениями. Обоснованность ответа. Необходимо мотивировать выбранный курс действий, приводящих к разрешению ситуации, и объяснить причины и рациональность его выбора.

Оценка «4» (хорошо) - выход из ситуации найден в целом верно, но с небольшими неточностями, имеются неточности в пояснении.

Оценка «3» (удовлетворительно) - выход из ситуации найден, верно, но не доведен до конца, либо в нем имеются ошибки, которые, однако, не приводят к принципиально неверному решению.

Оценка «2» (неудовлетворительно) - выход из ситуации не найден или найден неверно

Общая экзаменационная оценка выводиться по среднему баллу.