МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
Восточно-Сибирский государственный университет технологий и управления
Технологический колледж
Кафедра «Системы информатики»

СОГЛАСОВАНО:
Зам. директора по УМР ТК ВСГУТУ
_____ В.В. Пойдонова
«25» 04 2018 г.

УТВЕРЖДАЮ:
Директор ТК ВСГУТУ
_____С.Н.Сахаровский
«25» 04 2018 г.

РАБОЧАЯ ПРОГРАММА

к профессиональному модулю ПМ.01
«Разработка программных модулей программного обеспечения для компьютерных систем»

Специальность: 09.02.03 – Программирование в компьютерных системах

Форма обучения: Очная

Присваиваемая квалификация: Техник-программист

Улан-Удэ
2018

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 5B9C523B9C357D0B6A96448D522A58E73C9ED99B
Владелец: Сизов Игорь Геннадьевич
Действителен: с 30.12.2021 до 30.03.2023

Рабочая программа по профессиональному модулю ПМ.01 – «Разработка программных модулей программного обеспечения для компьютерных систем» разработана на кафедре «Системы информатики» ВСГУТУ и является частью программы подготовки специалистов среднего звена (ППССЗ), разработанной в соответствии с Федеральным государственным образовательным стандартом среднего профессионального образования по специальности: «09.02.03 «Программирование в компьютерных системах», утвержденным приказом Министерства образования и науки Российской Федерации от 28 июля 2014 г. №804.

Составители:

_____ Салтанов Р.А.

_____ Мердыгеев Б.Д.

Рабочая программа рассмотрена, обсуждена и одобрена на заседании кафедры «Системы информатики».
Протокол от «28» 03 2018 г.№ 9

Зав. кафедрой _____Михайлова С.С.

Аннотация к профессиональному модулю
ПМ.01 - «Разработка программных модулей программного обеспечения для компьютерных систем»

1. Цели и задачи изучения дисциплин профессионального модуля

Цель: формирование у обучающихся знаний, умений и практического опыта решения профессиональных задач, связанных с проектированием и разработкой алгоритмов решения таких задач, а также с их программной реализацией.
Задачи:

1) Получение знаний об основных этапах разработки программного обеспечения; основных принципов технологии структурного и объектно-ориентированного программирования; основных принципов отладки и тестирования программных продуктов; методах и средствах разработки технической документации;
2) Формирование умений осуществлять разработку кода программного модуля на современных языках программирования; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; оформлять документацию на программные средства; использовать инструментальные средства для автоматизации оформления документации;
3) Владеть практическим опытом разработки алгоритма поставленной задачи и реализации его средствами автоматизированного проектирования; разработки кода программного продукта на основе готовой спецификации на уровне модуля; использования инструментальных средств на этапе отладки программного продукта; проведения тестирования программного модуля по определенному сценарию.

2. Краткая характеристика профессионального модуля, его место в учебно-воспитательном процессе

Дисциплина формирует и развивает умения и практический опыт решения профессиональных задач, связанных с проектированием и разработкой алгоритмов решения таких задач, а также с их программной реализацией.

3. Место профессионального модуля в структуре образовательной программы

Профессиональный модуль ПМ.01 реализуется в трех семестрах на 2 и 3 году обучения. Трудоемкость освоения профессионального модуля (ПМ.01) составляет 870 часов.
Профессиональный модуль состоит из следующих элементов:

1. МДК.01.01 Системное программирование - 297 ч.;
2. МДК.01.02 Прикладное программирование – 249 ч.;
3. УП.01.01 Учебная практика – 252 ч.;
4. ПП.01.01 Производственная практика (по профилю специальности) – 72 ч.;
5. ПМ.01.ЭК Экзамен квалификационный;
6. Всего часов с учетом практики - 870 ч.

4. Взаимосвязь дисциплин профессионального модуля с предшествующими и последующими дисциплинами учебного плана подготовки

Формированию компетенций профессионального модуля ПМ.01 «Разработка программных модулей программного обеспечения компьютерных систем» способствуют компетенции, сформированные при изучении дисциплин «Основы программирования».

Компетенции, сформированные в результате освоения содержания дисциплин данного профессионального модуля, необходимы для освоения дисциплин профессионального модуля ПМ.02 «Разработка и администрирование баз данных».

5. Ожидаемые результаты освоения дисциплины

В результате освоения профессионального модуля, у обучающихся должны быть сформированы следующие компетенции:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Решать проблемы, оценивать риски и принимать решения в нестандартных ситуациях.

ОК 4. Осуществлять поиск, анализ и оценку информации, необходимой для постановки и решения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.

ОК 6. Работать в коллективе и команде, обеспечивать ее сплочение, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Быть готовым к смене технологий в профессиональной деятельности.

ПК 1.1 Выполнять разработку спецификаций отдельных компонент.

ПК 1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4 Выполнять тестирование программных модулей.

ПК 1.5 Осуществлять оптимизацию программного кода модуля.

ПК 1.6 Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.

ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

1. Цели и задачи дисциплины

1.1 Цели и задачи изучения дисциплины

Цель: формирование у обучающихся знаний, умений и практического опыта решения профессиональных задач, связанных с проектированием и разработкой алгоритмов решения таких задач, а также с их программной реализацией.

Задачи:

1) Получение знаний об основных этапах разработки программного обеспечения; основных принципов технологии структурного и объектно-ориентированного программирования; основных принципов отладки и тестирования программных продуктов; методах и средствах разработки технической документации;

2) Формирование умений осуществлять разработку кода программного модуля на современных языках программирования; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; оформлять документацию на программные средства; использовать инструментальные средства для автоматизации оформления документации;

3) Владеть практическим опытом разработки алгоритма поставленной задачи и реализации его средствами автоматизированного проектирования; разработки кода программного продукта на основе готовой спецификации на уровне модуля; использования инструментальных средств на этапе отладки программного продукта; проведения тестирования программного модуля по определенному сценарию.

2.1 Место дисциплины в структуре образовательной программы

Профессиональный модуль ПМ.01 реализуется в трех семестрах на 2 и 3 году обучения. Трудоемкость освоения профессионального модуля (ПМ.01) составляет 870 часов.

Профессиональный модуль состоит из следующих элементов:

1. МДК.01.01 Системное программирование - 297 ч.;
2. МДК.01.02 Прикладное программирование – 249 ч.;
3. УП.01.01 Учебная практика – 252 ч.;
4. ПП.01.01 Производственная практика (по профилю специальности) – 72 ч.;
5. ПМ.01.ЭК Экзамен квалификационный;
6. Всего часов с учетом практики - 870 ч.

Формированию компетенций профессионального модуля ПМ.01 «Разработка программных модулей программного обеспечения компьютерных систем» способствуют компетенции, сформированные при изучении дисциплин «Основы программирования».

Компетенции, сформированные в результате освоения содержания дисциплин данного профессионального модуля, необходимы для освоения дисциплин профессионального модуля ПМ.02 «Разработка и администрирование баз данных».

2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

2.1 Требования ФГОС СПО

В результате освоения профессионального модуля, у обучающихся должны быть сформированы следующие компетенции:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Решать проблемы, оценивать риски и принимать решения в нестандартных ситуациях.

ОК 4. Осуществлять поиск, анализ и оценку информации, необходимой для постановки и решения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.

ОК 6. Работать в коллективе и команде, обеспечивать ее сплочение, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Быть готовым к смене технологий в профессиональной деятельности.

ПК 1.1 Выполнять разработку спецификаций отдельных компонент.

ПК 1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4 Выполнять тестирование программных модулей.

ПК 1.5 Осуществлять оптимизацию программного кода модуля.

ПК 1.6 Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.

2.2 Планируемые результаты обучения

В результате совместной (с преподавателем) и индивидуальной (самостоятельной) деятельности в процессе изучения дисциплины обучающийся будет демонстрировать по освоению компетенций следующее:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Решать проблемы, оценивать риски и принимать решения в нестандартных ситуациях.

ОК 4. Осуществлять поиск, анализ и оценку информации, необходимой для постановки и решения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.

ОК 6. Работать в коллективе и команде, обеспечивать ее сплочение, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Быть готовым к смене технологий в профессиональной деятельности.

ПК 1.1 Выполнять разработку спецификаций отдельных компонент.

Иметь практический опыт: разработки спецификаций программных компонент;

Знать: основные принципы разработки компонент программных средств; основные принципы построения спецификации отдельных компонент; современные методы и инструментальные средства разработки спецификации;

Уметь: разрабатывать спецификации отдельных компонент программных средств;

ПК 1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

Иметь практический опыт: создания программ по разработанным алгоритмам для решения поставленных задач;

Знать: синтаксис и особенности конкретного языка программирования, на котором будет реализован модуль; основы алгоритмизации процессов;

Уметь: создавать программу по разработанному алгоритму как отдельный модуль;

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

Иметь практический опыт: работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок;

Знать: основные принципы отладки и тестирования программных продуктов;

Уметь: выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся;

ПК 1.4 Выполнять тестирование программных модулей.

Иметь практический опыт: проведения тестирования программы;

З**нать:** существующие методы тестирования программ; основные принципы проведения тестирования

Уметь: проводить тестирование программного модуля по определенному алгоритму;

ПК 1.5 Осуществлять оптимизацию программного кода модуля.

Иметь практический опыт: проведения оптимизации программы для повышения эффективности и читабельности кода;

Знать: методы оптимизации программного кода; основные способы сокращения объема кода программы и улучшения читабельности кода;

Уметь: использовать программные инструменты для уменьшения объема кода программы; проводить проверку программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы.

ПК 1.6 Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.

Иметь практический опыт: разработки документации программных средств и описания алгоритмов решения поставленных задач;

Знать: методы и средства разработки проектной и технической документации для описания программных средств и алгоритмов решения задач;

Уметь: создавать проектную и техническую документацию к программным средствам; описывать алгоритмы решения поставленных задач.

3. Объем дисциплины в часах – распределение учебного времени дисциплины

Содержание раздела выполнено в виде выписки из УП. В таблице 3.1 представлена информация по каждой форме обучения о распределении общей трудоемкости обучения в часах по семестрам, видов и объемов учебной работы в часах (лекции (Л)), лабораторные занятия (Лб), о распределении форм СРС – самостоятельной работы студентов, домашних заданий (ДЗ), индивидуальных заданий (ИЗ), курсовых проектов (КП), а также форм ПА – промежуточной аттестации студентов по учебной дисциплине (экзамен (Э), дифференцированный зачет (ДифЗ), зачет (З), другие формы контроля (ДФК)):

Таблица 3.1 – Распределение учебного времени дисциплины

Форма обучения	Семестр и его продолжительность (нед.)	РАСПРЕДЕЛЕНИЕ								Формы промежуточной аттестации
		Общей трудоемкости (час)	В том числе							
			На аудиторные занятия (час)				Консультации (час)	На СРС		
			Всего / в интерактивной форме	В том числе				(час)	Формы СРС	
				Л	Лб	КП				
1	2	3	4	5	6	7	8	9	10	11
Очная	МДК.01.01 – Системное программирование									
	4 семестр, 18 недель	160	108/30	72	36	-	-	52		Диф. зачет
	5 семестр 16 недель	137	96/10	32	32	32	1	40		Экзамен
	ИТОГО	**297**	**204/40**	**104**	**68**	**32**	**1**	**92**		
	МДК.01.02 – Прикладное программирование									
	5 семестр, 16 недель	96	64/13	32	32	-	-	32		Диф. зачет
	6 семестр 17 недель	153	102/20	51	51	-	2	49		Экзамен
	ИТОГО	**249**	**166/33**	**83**	**83**	**-**	**2**	**81**		
	УП.01.01 – Учебная практика									
	6 семестр	**252**								Диф. зачет
	ПП.01.01 – Производственная практика									
	8 семестр	**72**								Диф. зачет
	8 семестр	ПМ.01.ЭК – Квалификационный экзамен								Экзамен
	Всего часов с учетом практик	**870**								

4. Тематический план дисциплин профессионального модуля ПМ.01

4.1 Тематический план дисциплины МДК.01.01 – «Системное программирование»

Раздел 1: «Основы языка Python»

Тема 1 «Знакомство с языком программирования Python. Основы синтаксиса»

Основы синтаксиса языка Python. Правила оформления переменных, организация программного кода. Типы данных – числа, строки, булевы значения.

Тема 2 «Основы алгоритмизации. Линейные, условные, циклические алгоритмы»

Понятие алгоритма. Графическое описание алгоритма. Синтаксис последовательной программы на Python. Синтаксис циклического алгоритма. Синтаксис условного алгоритма. Совмещение различных алгоритмов в программе.

Тема 3 «Структуры данных. Массивы, кортежи, множества, словари»

Работа со структурами данных. Правила и синтаксис объявления структур данных в Python. Методы для работы со структурами данных. Алгоритмы на основе использования структур данных. Сортировка данных в массивах и словарях.

Тема 4 «Функции в Python. Конструкции обработки ошибок try/except»

Общее определение функции как части программы. Синтаксис объявления функции в Python. Особенности работы с функциями в Python. Общее определение ошибок программного кода. Синтаксис конструкции try/except. Описание возможных типов ошибок в Python и ситуации их возникновения.

Раздел 2: «Расширенное использование модулей и функций Python»

Тема 1 «Обзор встроенных методов Python для работы со строками.»

Основные способы взаимодействия со строками. Общие методы для работы с числами. Модуль NumPy. Методы модуля NumPy.

Тема 2 «Расширенные способы применения условных и циклических алгоритмов»

Способы применения сложных алгоритмов в решении практических задач. Общие паттерны программирования с использованием условных и циклических алгоритмов.

Тема 3 «Методы работы с массивами. Паттерны использования массивов в Python»

Особенности использования разных видов массивов Python. Шаблоны применения массивов в различных ситуациях.

Тема 4 «» Форматирование вывода в Python. Генерирование таблиц отчетов, оформление данных.

Использование методов форматирования вывода данных в консоль в Python. Перевод «сырых» данных в удобный для человеческого восприятия формат

Раздел 3: «Работа с файлами вPython»

Тема 1 «Работа с файлами. Чтение, запись, права доступа»

Принципы работы прав доступа к файлам в ОС Windows, ОС Linux. Файловые системы FAT32, NTFS, EXT. Методы для работы с файлами в Python. Чтение и запись файла в Python.

Тема 2 «Работа с файлами. Поиск файла папки по имени и маске»

Понятие типа файла. Понятие маски файла. Методы для поиска файла или папки в Python. Особенности поиска файлов в разных файловых системах.

Тема 3 «Работа с архивами. Принципы архивации файлов»

Понятие архива и архивации. Принципы архивации файлов. Методы архивации/разархивации файлов в Python. Использование архивов для хранения файлов.

Тема 4 «Создание, удаление, перемещение и удаление файла»

Методы для удаления и создания файлов в Python. Перемещение и копирование файлов с помощью модуля shutil.

Раздел 4: «Системное программирование на Python»

Тема 1 «Работа с системными процессами»

Понятие программного процесса. Системные и пользовательские процессы. Методы в Python для работы с процессами. Отслеживание, запуск и завершение процессов.

Тема 2 «Работа с пользователями»

Различия способов контроля учетных записей пользователей в ОС Windowsи ОС Linux. Политика безопасности системы. Методы в Python для работы с учетными записями пользователей.

Тема 3 «Работа с удаленными устройствами по протоколу SSH»

Принципы работы протокола SecureShell. Основы подключения и доступа к удаленному устройству по протоколу SSH. Методы в Python для работы с SHH. Удаленный доступ к файлам. Конфигурирование удаленного устройства по протоколу SSH

Раздел 5: «Объектно-ориентированное программирование на Python»

Тема 1 «Основные принципы ООП»

Объекты, абстракция, классификация. Декомпозиция объектов. Основные принципы ООП – инкапсуляция, наследование, полиморфизм.

Тема 2 «Классы и объекты»

Понятие класса и объекта как экземпляра класса. Члены класса – поля, свойства и методы. Обращение к членам класса.

Тема 3 «Инкапсуляция»

Спецификаторы доступа. Особенности доступа к данных классов в Python.

Тема 4 «Наследование»

Понятие родительского и дочернего классов. Механизм наследования классов в Python.

Тема 5 «Полиморфизм»

Конструкторы классов. Перегрузка методов. Переопределение методов и полей класса.

Раздел 6: «Построение графического интерфейса на Python»

Тема 1 «Знакомство с модулем tkinter»

Основы создания графического интерфейса с помощью модуля tkinter. Основные графические компоненты.

Тема 2 «Расположение элементов интерфейса»

Абсолютное и относительное позиционирование элементов. Менеджеры разметки place(), pack(), grid(). Использование панелей для создание сложных моделей интерфейсов.

Тема 3 «Модуль PIL. Вставка картинок в графический интерфейс»

Основы работы с модулем PIL. Загрузка иконки графического приложения. Вставка изображения в графический интерфейс. Изменение изображения внутри интерфейса

Тема 4 «Модуль winsound. Воспроизведение звуков в графическом интерфейсе»

Основы работы с модулем winsound. Способы воспроизведения звуков в интерфейсе. Поиск и загрузка звуков из свободного доступа.

4.1 Тематический план дисциплины МДК.01.02 – «Прикладное программирование»

Раздел 1: «Основы программирования»

Тема 1 «Алгоритмизация и введение в программирование»

Основные понятия. Алгоритм. Виды записи алгоритма. Базовые алгоритмические конструкции.

Тема 2 «Программирование на C\C++»

Краткая история языка С. Основная структура программы. Ключевые слова. Переменные и константы. Базовые типы данных. Базовые операции. Форматный ввод\вывод. Комментарии в программе.

Раздел 2: «Основы структурного программирования»

Тема 1 «Структурное программирование на C\C++»

Принципы структурного программирования. Базовые алгоритмические конструкции на языке C\C++. Ветвление. Циклы.

Тема 2 «Массивы»

Понятие массива. Элементы массива. Одномерные и многомерные массивы. Алгоритм сортировки пузырьком.

Тема 3 «Символьные строки»

Понятие символьной строки. Символьные константы. Специальные функции для работы со строками.

Тема 4 «Функции»

Объявление функции. Аргументы функции. Рекурсивные функции. Методы передачи массива в функцию. Области видимости функций и переменных.

Тема 5 «Указатели»

Понятие указателя. Операторы для работы с указателем. Операции с указателями. Ссылки. Взаимосвязь указателей и массивов. Способы передачи данных в функцию посредством указателей.

Тема 6 «Динамическое распределение памяти»

Распределение памяти в программе. Стэк и куча. Функции динамического распределения памяти. Выделение памяти под массивы.

Тема 7 «Пользовательские типы данных»

Понятие структуры. Понятие объединения. Понятие перечисления.

Тема 8 «Операции с разрядами»

Базовые системы счисления. Поразрядные операции. Битовые поля на основе структуры.

Тема 9 «Сложные структуры данных»

Динамическое выделение памяти под структуру. Типы механизмов доступа. Создание связанного списка на основе структуры.

Тема 10 «Файловый ввод\вывод»

Потоки и файлы. Стандартные потоки. Работа с файлами на основе функций языка С. Работа с файлами на основе функций языка С++.

Раздел 3: «Программирование с учетом внешних факторов»

Тема 1 «Исключения и ошибки»

Корректность и устойчивость программных систем. Исключения. Подходы к обработке ошибок и исключений. Механизм исключительных ситуаций. Обработка исключений.

Тема 2 «Директивы препроцессора»

Директивы препроцессора. Компоновка нескольких файлов в одну программу. Определение макросов. Директивы условной компиляции. Операторы препроцессора.

Раздел 4: «Объектно-ориентированное программирование»

Тема 1 «Основы объектно-ориентрованного программирования»

Основные понятия объектно-ориентрованного программирования. Понятие класса.

Тема 2 «Конструкторы и деструкторы»

Объявление конструктора. Виды конструкторов. Объявление деструктора. Статические элементы класса.

Тема 3 «Права доступа внутри класса»

Права доступа к компонентам класса. Дружественные функции и классы.

Тема 4 «Перегрузка операций»

Способы перегрузки операции. Особенности перегрузки глобальной операторной функции. Правила перегрузки операторов.

Тема 5 «Наследование»

Понятие наследования. Синтаксис наследования. Права доступа при наследовании. Общее и частное наследование. Иерархии классов. Множественное наследование.

Тема 6 «Абстрактные классы»

Полиморфизм включения и параметрический полиморфизм. Виртуальные функции, синтаксис. Абстрактные классы. Виртуальные деструкторы.

Тема 7 «Шаблоны классов»

Параметрический полиморфизм. Шаблоны в C++. Параметры шаблона. Шаблон функции. Вызов шаблонной функции. Специализация шаблонов. Выведение типов. Шаблон класса.

Тема 8 «Потоковые классы»

Иерархия потоков. Методы класса IOS. Манипуляторы. Стандартные потоковые объекты.

Тема 9 «Контейнерные классы»

Абстрактные структуры данных. Стек. Контейнеры.

Тема 10 «Итераторы и функциональные объекты»

Библиотека STL. Итераторы. Функциональные объекты. Библиотека algorithm.

5. Содержание дисциплин профессионального модуля ПМ.01

5.1 Содержание дисциплины МДК 01.01 «Системное программирование»

Таблица 5.1.1 - Лекционные занятия

Название разделов и тем	Количество часов
4 семестр	
Раздел 1: «Основы языка Python»	**18.0 ч.**
Тема 1 «Знакомство с языком программирования Python. Основы синтаксиса»	4.0 ч
Тема 2 «Основы алгоритмизации. Линейные, условные, циклические алгоритмы»	4.0 ч
Тема 3 «Структуры данных. Массивы, кортежи, множества, словари»	6.0 ч
Тема 4 «Функции в Python. Конструкции обработки ошибок try/except»	4.0 ч
Раздел 2: «Расширенное использование модулей и функций Python»	**18.0 ч.**
Тема 1 «Обзор встроенных методов Python для работы со строками»	4.0 ч
Тема 2 «Расширенные способы применения условных и циклических алгоритмов»	4.0 ч
Тема 3 «Методы работы с массивами. Паттерны использования массивов в Python»	6.0 ч
Тема 4 «» Форматирование вывода в Python. Генерирование таблиц отчетов, оформление данных.	4.0 ч
Раздел 3: «Работа с файлами в Python»	**18.0 ч.**
Тема 1 «Работа с файлами. Чтение, запись, права доступа»	4.0 ч
Тема 2 «Работа с файлами. Поиск файла папки по имени и маске»	6.0 ч
Тема 3 «Работа с архивами. Принципы архивации файлов»	4.0 ч
Тема 4 «Создание, удаление, перемещение и удаление файла»	4.0 ч
Раздел 4: «Системное программирование на Python»	**18.0 ч.**
Тема 1 «Работа с системными процессами»	6.0 ч
Тема 2 «Работа с пользователями»	4.0 ч
Тема 3 «Работа с удаленными устройствами по протоколу SSH»	8.0 ч
Итого 4 семестр	**72.0 ч**

5 семестр	
Название разделов и тем	**Количество часов**
Раздел 5: «Объектно-ориентированное программирование на Python»	**16.0 ч**
Тема 1 «Основные принципы ООП»	5.0 ч
Тема 2 «Классы и объекты»	5.0 ч
Тема 3 «Инкапсуляция»	2.0 ч
Тема 4 «Наследование»	2.0 ч
Тема 5 «Полиморфизм»	2.0 ч
Раздел 6: «Построение графического интерфейса на Python»	**16.0 ч**
Тема 1 «Знакомство с модулем tkinter»	4.0 ч
Тема 2 «Расположение элементов интерфейса»	4.0 ч
Тема 3 «Модуль PIL. Вставка картинок в графический интерфейс»	4.0 ч
Тема 4 «Модуль winsound. Воспроизведение звуков в графическом интерфейсе»	4.0 ч
Итого 5 семестр	**32.0 ч**
ИТОГО	**104 ч.**

Таблица 5.1.2 – Лабораторные занятия

Название разделов и тем	**Количество часов**
4 семестр	
Раздел 1: «Основы языка Python»	**8.0 ч.**
Тема 1 «Знакомство с языком программирования Python. Основы синтаксиса»	2.0 ч
Тема 2 «Основы алгоритмизации. Линейные, условные, циклические алгоритмы»	2.0 ч
Тема 3 «Структуры данных. Массивы, кортежи, множества, словари»	2.0 ч
Тема 4 «Функции в Python. Конструкции обработки ошибок try/except»	2.0 ч
Раздел 2: «Расширенное использование модулей и функций Python»	**8.0 ч.**
Тема 1 «Обзор встроенных методов Python для работы со строками»	2.0 ч
Тема 2 «Расширенные способы применения условных и циклических алгоритмов»	2.0 ч
Тема 3 «Методы работы с массивами. Паттерны использования массивов в Python»	2.0 ч
Тема 4 «» Форматирование вывода в Python. Генерирование таблиц отчетов, оформление данных.	2.0 ч
Раздел 3: «Работа с файлами вPython»	**8.0 ч.**
Тема 1 «Работа с файлами. Чтение, запись, права доступа»	2.0 ч
Тема 2 «Работа с файлами. Поиск файла папки по имени и маске»	2.0 ч
Тема 3 «Работа с архивами. Принципы архивации файлов»	2.0 ч
Тема 4 «Создание, удаление, перемещение и удаление файла»	2.0 ч

Название разделов и тем	Количество часов
Раздел 4: «Системное программирование на Python»	**12.0 ч.**
Тема 1 «Работа с системными процессами»	4.0 ч
Тема 2 «Работа с пользователями»	2.0 ч
Тема 3 «Работа с удаленными устройствами по протоколу SSH»	6.0 ч
Итого 4 семестр	**36.0 ч**
5 семестр	
Раздел 5: «Объектно-ориентированное программирование на Python»	**14.0 ч**
Тема 1 «Основные принципы ООП»	4.0 ч
Тема 2 «Классы и объекты»	4.0 ч
Тема 3 «Инкапсуляция»	2.0 ч
Тема 4 «Наследование»	2.0 ч
Тема 5 «Полиморфизм»	2.0 ч
Раздел 6: «Построение графического интерфейса на Python»	**18.0 ч**
Тема 1 «Знакомство с модулем tkinter»	6.0 ч
Тема 2 «Расположение элементов интерфейса»	6.0 ч
Тема 3 «Модуль PIL. Вставка картинок в графический интерфейс»	4.0 ч
Тема 4 «Модуль winsound. Воспроизведение звуков в графическом интерфейсе»	2.0 ч
Итого 5 семестр	**32.0 ч**
ИТОГО	**68 ч.**

Таблица 5.1.3 – Самостоятельная работа обучающихся

Название разделов и тем	Количество часов
4 семестр	
Раздел 1: «Основы языка Python»	**10.0 ч.**
Тема 1 «Знакомство с языком программирования Python. Основы синтаксиса»	2.0 ч
Тема 2 «Основы алгоритмизации. Линейные, условные, циклические алгоритмы»	2.0 ч
Тема 3 «Структуры данных. Массивы, кортежи, множества, словари»	4.0 ч
Тема 4 «Функции в Python. Конструкции обработки ошибок try/except»	2.0 ч
Раздел 2: «Расширенное использование модулей и функций Python»	**10.0 ч.**
Тема 1 «Обзор встроенных методов Python для работы со строками»	2.0 ч
Тема 2 «Расширенные способы применения условных и циклических алгоритмов»	2.0 ч
Тема 3 «Методы работы с массивами. Паттерны использования массивов в Python»	4.0 ч
Тема 4 «» Форматирование вывода в Python. Генерирование таблиц отчетов, оформление данных.	2.0 ч

Название разделов и тем	Количество часов
Раздел 3: «Работа с файлами в Python»	**10.0 ч.**
Тема 1 «Работа с файлами. Чтение, запись, права доступа»	2.0 ч
Тема 2 «Работа с файлами. Поиск файла папки по имени и маске»	2.0 ч
Тема 3 «Работа с архивами. Принципы архивации файлов»	4.0 ч
Тема 4 «Создание, удаление, перемещение и удаление файла»	2.0 ч
Раздел 4: «Системное программирование на Python»	**22.0 ч.**
Тема 1 «Работа с системными процессами»	6.0 ч
Тема 2 «Работа с пользователями»	4.0 ч
Тема 3 «Работа с удаленными устройствами по протоколу SSH»	12.0 ч
Итого 4 семестр	**52.0 ч**
5 семестр	
Раздел 5: «Объектно-ориентированное программирование на Python»	**18.0 ч**
Тема 1 «Основные принципы ООП»	6.0 ч
Тема 2 «Классы и объекты»	6.0 ч
Тема 3 «Инкапсуляция»	2.0 ч
Тема 4 «Наследование»	2.0 ч
Тема 5 «Полиморфизм»	2.0 ч
Раздел 6: «Построение графического интерфейса на Python»	**22.0 ч**
Тема 1 «Знакомство с модулем tkinter»	8.0 ч
Тема 2 «Расположение элементов интерфейса»	6.0 ч
Тема 3 «Модуль PIL. Вставка картинок в графический интерфейс»	4.0 ч
Тема 4 «Модуль winsound. Воспроизведение звуков в графическом интерфейсе»	4.0 ч
Итого 5 семестр	**40.0 ч**
ИТОГО	**92 ч.**

Таблица 5.1.4 – Учебно-методическая (технологическая) карта СРС по очной форме

Общие затраты времени по всем видам СРС	Количество часов
4 семестр	
Проработка материалов лекций (подготовка к лекционным занятиям)	14 ч.
Подготовка к лабораторно-практическим занятиям	30 ч.
Подготовка к промежуточной аттестации (диф.зачет)	8 ч.
Итого за 4 семестр:	**52 ч.**
5 семестр	
Проработка материалов лекций (подготовка к лекционным занятиям)	4
Подготовка к лабораторно-практическим занятиям	28
Подготовка к промежуточной аттестации (экзамен)	8
Итого за 5 семестр:	**40 ч.**
ИТОГО	**92 ч.**

5.1 Содержание дисциплины МДК 01.02 «Прикладное программирование»

Таблица 5.1.1 - Лекционные занятия

Название разделов и тем	Количество часов
5 семестр	
Раздел 1: «Основы программирования»	**4**
Тема 1 «Алгоритмизация и введение в программирование»	2
Тема 2 «Программирование на C\C++»	2
Раздел 2: «Основы структурного программирования»	**28**
Тема 1 «Структурное программирование на C\C++»	4
Тема 2 «Массивы»	2
Тема 3 «Символьные строки»	4
Тема 4 «Функции»	2
Тема 5 «Указатели»	2
Тема 6 «Динамическое распределение памяти»	2
Тема 7 «Пользовательские типы данных»	4
Тема 8 «Операции с разрядами»	2
Тема 9 «Сложные структуры данных»	2
Тема 10 «Файловый ввод\вывод»	4
Итого за 5 семестр	**32 ч.**
6 семестр	
Раздел 3: «Программирование с учетом внешних факторов»	**8**
Тема 1 «Исключения и ошибки»	4
Тема 2 «Директивы препроцессора»	4
Раздел 4: «Объектно-ориентированное программирование»	**43**
Тема 1 «Основы объектно-ориентрованного программирования»	5
Тема 2 «Конструкторы и деструкторы»	4
Тема 3 «Права доступа внутри класса»	4
Тема 4 «Перегрузка операций»	4
Тема 5 «Наследование»	5
Тема 6 «Абстрактные классы»	5
Тема 7 «Шаблоны классов»	4
Тема 8 «Потоковые классы»	4
Тема 9 «Контейнерные классы»	4
Тема 10 «Итераторы и функциональные объекты»	4
Итого 6 семестр	**51**
ИТОГО	**83**

Таблица 5.1.2 – Лабораторные занятия

Название разделов и тем	Количество часов
5 семестр	
Раздел 1: «Основы программирования»	**4**
Тема 1 «Алгоритмизация и введение в программирование»	2
Тема 2 «Программирование на C\C++»	2

Название разделов и тем	Количество часов
Раздел 2: «Основы структурного программирования»	**28**
Тема 1 «Структурное программирование на C\C++»	4
Тема 2 «Массивы»	2
Тема 3 «Символьные строки»	4
Тема 4 «Функции»	2
Тема 5 «Указатели»	2
Тема 6 «Динамическое распределение памяти»	2
Тема 7 «Пользовательские типы данных»	4
Тема 8 «Операции с разрядами»	2
Тема 9 «Сложные структуры данных»	2
Тема 10 «Файловый ввод\вывод»	4
Итого за 5 семестр	**32 ч.**
6 семестр	
Раздел 3: «Программирование с учетом внешних факторов»	**8**
Тема 1 «Исключения и ошибки»	4
Тема 2 «Директивы препроцессора»	4
Раздел 4: «Объектно-ориентированное программирование»	**43**
Тема 1 «Основы объектно-ориентрованного программирования»	5
Тема 2 «Конструкторы и деструкторы»	4
Тема 3 «Права доступа внутри класса»	4
Тема 4 «Перегрузка операций»	4
Тема 5 «Наследование»	5
Тема 6 «Абстрактные классы»	5
Тема 7 «Шаблоны классов»	4
Тема 8 «Потоковые классы»	4
Тема 9 «Контейнерные классы»	4
Тема 10 «Итераторы и функциональные объекты»	4
Итого 6 семестр	**51**
ИТОГО	**83**

Таблица 5.1.3 – Самостоятельная работа обучающихся

Название разделов и тем	Количество часов
5 семестр	
Раздел 1: «Основы программирования»	**4**
Тема 1 «Алгоритмизация и введение в программирование»	2
Тема 2 «Программирование на C\C++»	2
Раздел 2: «Основы структурного программирования»	**28**
Тема 1 «Структурное программирование на C\C++»	4
Тема 2 «Массивы»	2
Тема 3 «Символьные строки»	4
Тема 4 «Функции»	2
Тема 5 «Указатели»	2
Тема 6 «Динамическое распределение памяти»	2
Тема 7 «Пользовательские типы данных»	4
Тема 8 «Операции с разрядами»	2
Тема 9 «Сложные структуры данных»	2
Тема 10 «Файловый ввод\вывод»	4
Итого за 5 семестр	**32 ч.**

Название разделов и тем	Количество часов
6 семестр	
Раздел 3: «Программирование с учетом внешних факторов»	**7**
Тема 1 «Исключения и ошибки»	4
Тема 2 «Директивы препроцессора»	3
Раздел 4: «Объектно-ориентированное программирование»	**42**
Тема 1 «Основы объектно-ориентрованного программирования»	4
Тема 2 «Конструкторы и деструкторы»	4
Тема 3 «Права доступа внутри класса»	4
Тема 4 «Перегрузка операций»	4
Тема 5 «Наследование»	5
Тема 6 «Абстрактные классы»	5
Тема 7 «Шаблоны классов»	4
Тема 8 «Потоковые классы»	4
Тема 9 «Контейнерные классы»	4
Тема 10 «Итераторы и функциональные объекты»	4
Итого 6 семестр	**49**
ИТОГО	**81**

Таблица 5.1.4 – Учебно-методическая (технологическая) карта СРС по очной форме

Общие затраты времени по всем видам СРС	Количество часов
5 семестр	
Проработка материалов лекций (подготовка к лекционным занятиям)	10 ч.
Подготовка к лабораторно-практическим занятиям	18 ч.
Подготовка к промежуточной аттестации (диф.зачет)	4 ч.
Итого за 4 семестр:	**32 ч.**
6 семестр	
Проработка материалов лекций (подготовка к лекционным занятиям)	12
Подготовка к лабораторно-практическим занятиям	28
Подготовка к промежуточной аттестации (экзамен)	9
Итого за 5 семестр:	**49 ч.**
ИТОГО	**81ч.**

6. Учебно-методическое и информационное обеспечение дисциплины (по видам учебной дисциплины и формам контроля)

Таблица 6.1 – Учебно-методическое и информационное обеспечение дисциплин профессионального модуля ПМ.01

№ п/п	Наименование учебно-методического материала
Перечень основной учебной литературы	
1	Михайлов, Ю. Ф., Зайцева, С. А., Козлов, О. А. Системное программное обеспечение. [Электронный ресурс] – Шуя: ФГБОУ ВПО "ШГПУ", 2013. 79 с. Режим доступа: https://api.rucont.ru/api/efd/reader?file=206357 ЭБС Руконт.
2	Федорова, Г. Н. Разработка модулей программного обеспечения для компьютерных систем : учебник для среднего профессионального образования - 2-е изд., стер. - Москва : Издательский центр "Академия", 2018. – 383 стр. – 10 экз.
3	Сорокин, А. А. Объектно-ориентированное программирование : учебное пособие (курс лекций).— Ставрополь : изд-во СКФУ, 2014 .— 174 с. Режим доступа: https://api.rucont.ru/api/efd/reader?file=304186
4	Федорова, Г. Н. Разработка модулей программного обеспечения для компьютерных систем : учебник для среднего профессионального образования - 2-е изд., стер. - Москва : Издательский центр "Академия", 2018. – 383 стр. – 10 экз.
Перечень дополнительной литературы	
5	Дель, Л. А., Уйманова, Н. А. Системное программирование: [Электронный ресурс] метод. указания к лаб. работам - Оренбургский гос. ун-т, Оренбург : ОГУ, 2013. – 63 с. Режим доступа: https://api.rucont.ru/api/efd/reader?file=227491 ЭБС Руконт.
6	Николаев, Е. И. Объектно-ориентированное программирование : учебное пособие. Профиль подготовки «Прикладное программирование в информационных системах» — Ставрополь : изд-во СКФУ, 2015 .— 225 с. Режим доступа: https://api.rucont.ru/api/efd/reader?file=578811
7	Дель, Л. А, Уйманова, Н. А. Прикладное программирование : метод. указания к лаб. работам - Оренбургский гос. ун-т.— Оренбург : ОГУ, 2013 .— 90 с. : ил. Режим доступа: https://api.rucont.ru/api/efd/reader?file=227432
Методические указания для обучающихся (МУ)	
1	«Методические указания к выполнению лабораторных работ по дисциплине «Системное программирование» размещены по следующему адресу https://www.esstu.ru/uportal/document/list.htm?departmentId=57&categoryId=4274
2	«Методические указания к выполнению лабораторных работ по дисциплине «Прикладное программирование» размещены по следующему адресу https://www.esstu.ru/uportal/document/list.htm?departmentId=57&categoryId=4274
Лицензионное программное обеспечение	
1	Microsoft Windows 7. Акт на передачу прав №533 от 11.11.2016 на ПО Microsoft Imagine premium electronic software сроком действия – три года
2	Microsoft Office 2010 – Open License 62024856, бессрочная лицензия
3	Python 3.7.1 – Свободно-распространяемое ПО, лицензия Python Software Foundation License
4	PyCharm Community Edition – Свободно-распространяемое ПО, лицензия Apache 2
5	Microsoft Visual Studio 2010 - Акт на передачу прав №533 от 11.11.2016 на ПО Microsoft Imagine premium electronic software сроком действия - три года
Современные профессиональные базы данных и информационно-справочные системы	
ГОСТ-эксперт, единая база ГОСТов РФ: http://gostexpert.ru	
Периодические издания	
1	Теоретический и прикладной научно-технический журнал «Программная инженерия». Режим доступа: http://novtex.ru/prin/rus/.
2	Вестник Южно-Уральского государственного университета. Серия "Компьютерные

	технологии, управление, радиоэлектроника". Режим доступа: https://rucont.ru/efd/13323.
3	Международный научно-практический журнал «Программные продукты и системы». Режим доступа: http://www.swsys.ru/.
Нормативные документы (на кафедре)	
1	Положение о компьютерных классах кафедры «Системы информатики»

7. Материально-техническое обеспечение дисциплины

Таблица 7.1 – Материально-технические ресурсы, используемые в дисциплинах профессионального модуля ПМ.01

Используемые специализированные аудитории и лаборатории		Перечень оборудования и систем		Примечание
№	Наименование	№	Наименование Кол.	
1	Учебная аудитория для проведения занятий лекционного типа	1 2 3	Специализированная мебель Учебная доска Наборы демонстрационного оборудования и учебно-наглядных пособий	Microsoft Office 2010; Windows 7; Антивирус.
2	Учебная аудитория для проведения занятий семинарского типа	1 2 3 4 5	Специализированная мебель Учебная доска Компьютерная техника Подключение к сети «Интернет» Доступ в электронную информационно-образовательную среду	Microsoft Office 2010; Windows 7; Антивирус; Python 3.7.1; PyCharm Community Edition; Microsoft Visual Studio 2010
3	Учебная аудитория для курсового проектирования (выполнения курсовых работ)	1 2 3	Специализированная мебель Учебная доска Компьютерная техника	Microsoft Office 2010; Windows 7; Антивирус; Python 3.7.1; PyCharm Community Edition; Microsoft Visual Studio 2010
4	Учебная аудитория для групповых и индивидуальных консультаций	1 2 3	Специализированная мебель Учебная доска Компьютерная техника	Microsoft Office 2010; Windows 7; Антивирус; Python 3.7.1; PyCharm Community Edition; Microsoft Visual Studio 2010
5	Учебная аудитория для текущего контроля и промежуточной аттестации	1 2	Специализированная мебель Учебная доска	Microsoft Office 2010; Windows 7; Антивирус; Python 3.7.1; PyCharm Community Edition; Microsoft Visual Studio 2010
6	Помещение для самостоятельной работы	1 2 3 4 5	Специализированная мебель Учебная доска Компьютерная техника Подключение к сети «Интернет» Доступ в электронную информационно-образовательную среду	Microsoft Office 2010; Windows 7; Антивирус; Python 3.7.1; PyCharm Community Edition; Microsoft Visual Studio 2010
7	Помещение для хранения и профилактического обслуживания учебного оборудования	1 2 3	Стеллажи Специализированная мебель Набор для диагностики сети и компьютеров	

8. Форма и методика проведения текущего контроля успеваемости и промежуточной аттестации студентов

Формы и методы контроля и оценки результатов обучения, позволяющие проверять у обучающихся профессиональные компетенции, представлены в таблице 8.1

Таблица 8.1 – Формы и методы контроля и оценки профессиональных компетенций

Результаты (освоенные профессиональные компетенции)	**Основные показатели оценки результата**	**Формы и методы контроля и оценки**
ПК 1.1 Выполнять разработку спецификаций отдельных компонент.	Разработанные спецификации программных компонентов операционной системы	*Экспертная оценка выполненных лабораторных работ, курсового проекта, частей ВКР*
ПК 1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.	Написанная на языке программирования высокого уровня в виде отдельного модуля программа, разработанная с учетом составленной ранее спецификации	*Экспертная оценка выполненных лабораторных работ, курсового проекта, частей ВКР*
Результаты (освоенные профессиональные компетенции)	**Основные показатели оценки результата**	**Формы и методы контроля и оценки**
ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.	Результаты отладки программного кода с использованием встроенного в среду пошагового дебаггера	*Экспертная оценка выполненных лабораторных работ, работ во время прохождения практики и выполнения ВКР*
ПК 1.4 Выполнять тестирование программных модулей.	Разработанный unit-тест для программного модуля с успешно проведенным тестированием на всей тестовой выборке	*Экспертная оценка выполненных лабораторных работ, работ во время прохождения практики и выполнения ВКР*
ПК 1.5 Осуществлять оптимизацию программного кода модуля.	Результаты оптимизации разработанного программного кода модуля в виде повышения скорости выполнения работы программы (если это возможно) либо сокращения объема кода при сохранении предусмотренного спецификацией функционала программы	*Экспертная оценка выполненных лабораторных работ, работ во время прохождения практики и выполнения ВКР*
ПК 1.6 Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.	Корректно составленная документация в форме встроенных в программу технических комментариев с описанием и пользовательской инструкции	*Экспертная оценка выполненных лабораторных работ, работ во время прохождения практики и выполнения ВКР*

Формы и методы контроля и оценки результатов обучения, позволяющие проверять у обучающихся развитие общих компетенций, представлены в таблице 8.2.

Таблица 8.2 – Формы и методы контроля и оценки развития общих компетенций

Результаты (освоенные общие компетенции)	**Основные показатели оценки результата**	**Формы и методы контроля и оценки**
ОК1 Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.	Успешное освоение теоретической и практической части ОПОП	*Зачет, защита лабораторных работ*
ОК2 Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.	Выполнение курсового проекта по разработке и реализации базы данных и программного приложения для выбранной предметной области, ВКР	*Защита курсового проекта, производственной практики, успешная защита ВКР*
ОК3 Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	Анализ предметной области с целью выбора объекта автоматизации в период прохождения производственной практики	*Защита производственных практик 1, 2*
ОК4 Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.	Найденная информация, дающая исчерпывающее представление о выбранной предметной области с целью выполнения курсового проекта по разработке и реализации информационной системы	*Экспертная оценка раздела курсового проекта «Анализ предметной области и разработка концептуальной модели предметной области»*
ОК5 Использовать информационно-коммуникационные технологии в профессиональной деятельности.	Форматирование и оформление текстовых, графических документов, выполнение расчетов в электронных таблицах и представление числовой информации в виде диаграмм, графиков с целью анализа информации.	*Владение ИКТ на уровне пользователя ПК, зачет по дисциплине «Выполнение работ оператора»*
ОК6 Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.	Работа в коллективе в период производственных практик.	*Положительная характеристика и отзывы с производственной практики*
ОК7 - Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий	Работа в малых группах при выполнении лабораторных работ, решении ситуационных задач на интерактивных занятиях.	*Взаимная экспертная оценка работы в малых группах на лабораторных и интерактивных занятиях*

Результаты (освоенные общие компетенции)	Основные показатели оценки результата	Формы и методы контроля и оценки
ОК8 - Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации	Организация самостоятельных занятий при изучении профессионального модуля	*Интерпретация результатов наблюдений за деятельностью обучающегося в процессе освоения образовательной программы*
ОК9 - Быть готовым к смене технологий в профессиональной деятельности	Анализ инноваций в области информационных технологий	*Интерпретация результатов наблюдений за деятельностью обучающегося в процессе освоения образовательной программы*

Для оценки уровня учебных достижений обучающихся принята пятибальная шкала. Шкала оценки представлена в таблице 8.3.

Таблица 8.3 – Шкала оценки

Итоговая оценка по дисциплине			
Неудовлетворительно	Удовлетворительно	Хорошо	Отлично 5
0 – 2.9	3.0-3.49	3.5-4.4	4.5-5.0

В таблице 8.4 и 8.5 представлены контрольные мероприятия по видам учебной деятельности для дисциплин МДК.01.01 и МДК.01.02 соответственно.

Таблица 8.5 – Контрольные мероприятия по дисциплине МДК.01.01 – Инфокоммуникационные системы и сети

№	Наименование разделов	Форма оценки теоретической составляющей	Форма оценки практической составляющей
1	**4 семестр**	Промежуточная аттестация	Защита лабораторных работ
Промежуточная аттестация – дифференцированный зачет			
2	**5 семестр**	Промежуточная аттестация	Защита лабораторных работ Защита курсового проекта
Промежуточная аттестация – экзамен			

Таблица 8.5 – Контрольные мероприятия по дисциплине МДК.01.02 – Технология разработки и защиты баз данных

№	Наименование разделов	Форма оценки теоретической составляющей	Форма оценки практической составляющей
2	**5 семестр**	Промежуточная аттестация	Защита лабораторных работ
Промежуточная аттестация – дифференцированный зачет			
3	**6 семестр**	Промежуточная аттестация	Защита лабораторных работ
Промежуточная аттестация – экзамен			

Контрольно-измерительные материалы дисциплины представлены в разделе «Фонд оценочных средств».

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Восточно-Сибирский государственный университет технологий управления»

Технологический колледж
Кафедра «Системы информатики»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

к дисциплине МДК.01.01 «Системное программирование»

Специальность: 09.02.03 - Программирование в компьютерных системах

Форма обучения: Очная

Присваиваемая квалификация: Техник-программист

Улан-Удэ
2018

Паспорт
фонда оценочных средств
по дисциплине МДК.01.01 «Системное программирование»

1. ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

В результате освоения дисциплины у обучающихся должны быть сформированы следующие компетенции:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Решать проблемы, оценивать риски и принимать решения в нестандартных ситуациях.

ОК 4. Осуществлять поиск, анализ и оценку информации, необходимой для постановки и решения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.

ОК 6. Работать в коллективе и команде, обеспечивать ее сплочение, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Быть готовым к смене технологий в профессиональной деятельности.

ПК 1.1 Выполнять разработку спецификаций отдельных компонент.

ПК 1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4 Выполнять тестирование программных модулей.

ПК 1.5 Осуществлять оптимизацию программного кода модуля.

ПК 1.6 Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.

2. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ НА РАЗЛИЧНЫХ ЭТАПАХ ИХ ФОРМИРОВАНИЯ, ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ

2.1. Описание показателей оценивания компетенций

Фонд оценочных средств сформирован на основе ключевых принципов оценивания:

- валидность: объекты оценки должны соответствовать поставленным целям обучения;
- надежность: использование единообразных показателей и критериев для оценивания достижений;
- объективность: получение объективных и достоверных результатов при проведении контроля с различными целями.

В процессе оценивания знаний, умений и навыков обучающийся должен демонстрировать следующие результаты по освоению компетенций:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

Знать: как и где можно применить теоретические знания и практические умения, полученные в результате освоения дисциплины в различных сферах деятельности.

ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

Уметь: применять теоретические знания и практические умения для выполнения индивидуальных заданий, курсовых проектов, заданий учебной и производственной практик и выпускной квалификационной работы.

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

Уметь: самостоятельно анализировать предметную область при выборе объекта автоматизации в период прохождения производственной практики.

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

Уметь: находить информацию, дающую исчерпывающее представление о той предметной области, для которой разрабывается база данных и программное приложение в рамках курсового проектирования или выпускной квалификационной работы

ОК 5. Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.

Уметь: выполнять форматирование и оформление текстовых, графических документов, расчеты в электронных таблицах и представление числовой информации в виде диаграмм и графиков для анализа данных.

ОК 6. Работать в коллективе и команде, обеспечивать ее сплочение, эффективно общаться с коллегами, руководством, потребителями.

Уметь: строить рабочие и коммуникативные отношения в период учебной и производственной практик.

ОК 7. Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий.

Уметь: строить рабочие и коммуникативные отношения в малых группах при выполнении лабораторных работ, решения ситуационных задач на интерактивных занятиях.

ОК 8. Самостоятельно определять задачи, заниматься самообразованием, осознанно планировать повышение квалификации.

Уметь: оценивать собственный уровень профессионального и личностного развития, определять пути самообразования.

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

Уметь: определять новые технологии и быть готовым к их освоению и использованию в профессиональной деятельности.

ПК 1.1 Выполнять разработку спецификаций отдельных компонент.

Знать: основные положения теории разработки системных компонентов; основные принципы построения концептуальной, логической и физической модели данных; современные инструментальные средства разработки схемы процессов; модели и структуры информационных систем;

Уметь: работать с современными case-средствами проектирования моделей данных;

ПК 1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

Знать: синтаксис и особенности конкретного языка программирования, на котором будет реализован модуль; основы алгоритмизации процессов;

Уметь: создавать программу по разработанному алгоритму как отдельный модуль;

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

Знать: основные принципы отладки и тестирования программных продуктов;

Уметь: выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся;

ПК 1.4 Выполнять тестирование программных модулей.

Знать: существующие типы тестирования программ; основные принципы проведения unit-тестирования

Уметь: проводить тестирование программного модуля по определенному сценарию

ПК 1.5 Осуществлять оптимизацию программного кода модуля.

Знать: паттерны оптимизации программного кода; основные способы сокращения объема кода программы

Уметь: использовать программные инструменты для уменьшения объема кода программы; проводить ревью программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы.

ПК 1.6 Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.

Знать: методы и средства разработки технической документации;

Уметь: оформлять документацию на программные средства; использовать инструментальные средства для автоматизации оформления документации;

Описание показателей, форм и методов оценивания компетенций представлены в таблице 2.1, в которой отображены процессы формирования вклада дисциплины «Системное программирование» в формирование компетенций с ПК1.1 по ПК1.6 и с ОК-1 по ОК-9.

Таблица 2.1. Показатели оценивания компетенций на различных этапах их формирования

№ п/п	Контролируемые разделы (темы) дисциплины*	Код контролируемой компетенции **	Наименование оценочного средства
1	**Раздел 1.** Основы языка Python	ОК1, ПК 1.1, ПК 1.2	Вопросы по лабораторной работе
2	**Раздел 2.** Расширенное использование модулей и функций Python	ОК2, ОК3, ПК 1.1, ПК 1.2	Вопросы по лабораторной работе
3	**Раздел 3.** Работа с файлами в Python	ПК 1.1, ПК 1.2, ПК 1.3	Вопросы по лабораторной работе
4	**Раздел 4.** Системное программирование на Python	ОК5, ОК6, ПК 1.3, ПК 1.4,	Вопросы по лабораторной работе
5	**Раздел 5.** Объектно-ориентированное программирование на Python	ОК4, ОК7, ПК 1.4, ПК 1.5	Вопросы по лабораторной работе Курсовой проект
6	**Раздел 6.** Построение графического интерфейса на Python	ОК 8, ОК 9, ПК 1.5, ПК 1,6	Вопросы по лабораторной работе Курсовой проект

2.2. Критерии оценивания компетенций, описание шкал оценивания

Описание критериев и шкал оценивания компетенций ПК-1.1, ПК-1.2, ПК-1.3, ПК-1.4, ПК-1.5, ПК-1.6 в рамках оценивания результатов обучения по профессиональному модулю «Разработка и администрирование баз данных» приведено в таблицах 2.2.1, 2.2.2, 2.2.3, 2.2.4, 2.2.5, 2.2.6 соответственно.

Таблица 2.2.1 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.1

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание основных принципов разработки компонент программных средств; основных принципов построения спецификации отдельных компонент; современных методов и инструментальные средства разработки спецификации. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание основных принципов разработки компонент программных средств; основных принципов построения спецификации отдельных компонент; современных методов и инструментальные средства разработки спецификации.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание основных принципов разработки компонент программных средств; основных принципов построения спецификации отдельных компонент; современных методов и инструментальные средства разработки спецификации.	Сформированные систематические знания, глубокое понимание основных принципов разработки компонент программных средств; основных принципов построения спецификации отдельных компонент; современных методов и инструментальные средства разработки спецификации.
	Отсутствие сформированных умений / частично освоенные умения разрабатывать спецификации отдельных компонент программных средств.	В основном сформированные умения разрабатывать спецификации отдельных компонент программных средств. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения разрабатывать спецификации отдельных компонент программных средств. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения разрабатывать спецификации отдельных компонент программных средств.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта разработки спецификаций программных компонент	В основном сформированные навыки при приобретении практического опыта разработки спецификаций программных компонент. В целом успешные навыки работы, но применяемые не систематически	Сформированные навыки при приобретении практического опыта разработки спецификаций программных компонент. В целом успешное применение опыта работы, но содержащее отдельные пробелы	Полностью приобретенный опыт разработки спецификаций программных компонент. Успешное и систематическое применение навыков работы

Таблица 2.2.2 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.2

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание синтаксиса и особенностей конкретного языка программирования, на котором будет реализован модуль; основ алгоритмизации процессов. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание синтаксиса и особенностей конкретного языка программирования, на котором будет реализован модуль; основ алгоритмизации процессов.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание синтаксиса и особенностей конкретного языка программирования, на котором будет реализован модуль; основ алгоритмизации процессов.	Сформированные систематические знания, глубокое понимание синтаксиса и особенностей конкретного языка программирования, на котором будет реализован модуль; основ алгоритмизации процессов.
	Отсутствие сформированных умений / частично освоенные умения создавать программу по разработанному алгоритму как отдельный модуль.	В основном сформированные умения создавать программу по разработанному алгоритму как отдельный модуль. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения создавать программу по разработанному алгоритму как отдельный модуль. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения создавать программу по разработанному алгоритму как отдельный модуль.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта создания программ по разработанным алгоритмам для решения поставленных задач.	В основном сформированные навыки при приобретении практического опыта создания программ по разработанным алгоритмам для решения поставленных задач. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта создания программ по разработанным алгоритмам для решения поставленных задач. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт создания программ по разработанным алгоритмам для решения поставленных задач. Успешное и систематическое применение навыков работы.

Таблица 2.2.3 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.3

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание основных принципов отладки и тестирования программных продуктов. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание основных принципов отладки и тестирования программных продуктов.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание основных принципов отладки и тестирования программных продуктов.	Сформированные систематические знания, глубокое понимание основных принципов отладки и тестирования программных продуктов.
	Отсутствие сформированных умений / частично освоенные умения выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся.	В основном сформированные умения выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок.	В основном сформированные навыки при приобретении практического опыта работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок. Успешное и систематическое применение навыков работы.

Таблица 2.2.4 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.4

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание существующих методов тестирования программ; основных принципов проведения тестирования. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание существующих методов тестирования программ; основных принципов проведения тестирования.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание существующих методов тестирования программ; основных принципов проведения тестирования.	Сформированные систематические знания, глубокое понимание существующих методов тестирования программ; основных принципов проведения тестирования.
	Отсутствие сформированных умений / частично освоенные умения проведения тестирование программного модуля по определенному алгоритму.	В основном сформированные умения проведения тестирование программного модуля по определенному алгоритму. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения проведения тестирование программного модуля по определенному алгоритму. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения проведения тестирование программного модуля по определенному алгоритму.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта проведения тестирования программы.	В основном сформированные навыки при приобретении практического опыта проведения тестирования программы. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта проведения тестирования программы. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт проведения тестирования программы. Успешное и систематическое применение навыков работы.

Таблица 2.2.5 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.5

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание методов оптимизации программного кода; основных способов сокращения объема кода программы и улучшения читабельности кода. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание методов оптимизации программного кода; основных способов сокращения объема кода программы и улучшения читабельности кода.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание методов оптимизации программного кода; основных способов сокращения объема кода программы и улучшения читабельности кода.	Сформированные систематические знания, глубокое понимание методов оптимизации программного кода; основных способов сокращения объема кода программы и улучшения читабельности кода.
	Отсутствие сформированных умений / частично освоенные умения использования программные инструменты для уменьшения объема кода программы; проведения проверки программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы.	В основном сформированные умения использования программные инструменты для уменьшения объема кода программы; проведения проверки программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения использования программные инструменты для уменьшения объема кода программы; проведения проверки программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения использования программные инструменты для уменьшения объема кода программы; проведения проверки программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта проведения оптимизации программы для повышения эффективности и читабельности кода.	В основном сформированные навыки при приобретении практического опыта проведения оптимизации программы для повышения эффективности и читабельности кода. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта проведения оптимизации программы для повышения эффективности и читабельности кода. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт проведения оптимизации программы для повышения эффективности и читабельности кода. Успешное и систематическое применение навыков работы.

Таблица 2.2.6 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.6

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание методов и средств разработки проектной и технической документации для описания программных средств и алгоритмов решения задач. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание методов и средств разработки проектной и технической документации для описания программных средств и алгоритмов решения задач.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание методов и средств разработки проектной и технической документации для описания программных средств и алгоритмов решения задач.	Сформированные систематические знания, глубокое понимание методов и средств разработки проектной и технической документации для описания программных средств и алгоритмов решения задач.
	Отсутствие сформированных умений / частично освоенные умения создавать проектную и техническую документацию к программным средствам; описывать алгоритмы решения поставленных задач.	В основном сформированные умения создавать проектную и техническую документацию к программным средствам; описывать алгоритмы решения поставленных задач. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения создавать проектную и техническую документацию к программным средствам; описывать алгоритмы решения поставленных задач. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения создавать проектную и техническую документацию к программным средствам; описывать алгоритмы решения поставленных задач.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта разработки документации программных средств и описания алгоритмов решения поставленных задач.	В основном сформированные навыки при приобретении практического опыта разработки документации программных средств и описания алгоритмов решения поставленных задач. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта разработки документации программных средств и описания алгоритмов решения поставленных задач. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт разработки документации программных средств и описания алгоритмов решения поставленных задач. Успешное и систематическое применение навыков работы.

3. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ И МАТЕРИАЛЫ, НЕОБХОДИМЫЕ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ И НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИИ

3.1 Типовые контрольные задания и материалы по дисциплине МДК.01.01 «Системное программирование»

3.1.2 Лабораторные работы

Семестр 4

Лабораторная работа 1. «Введение в типы объектов языка Python»
Содержание:

1. Цель и задачи лабораторной работы №1
2. Формулирование предметной области задачи
3. Знакомство с типами данных в Python
4. Создание проекта программного приложения
5. Написание программного кода по заданному алгоритму
6. Запуск и тестирование результатов работы программы

Контрольные вопросы

1. Назовите четыре базовых типа данных в языке Python.
2. Почему они называются базовыми?
3. Что означает термин «неизменяемый» и какие три базовых типа языка Python являются неизменяемыми?

Лабораторная работа 2. «Структуры данных – списки, словари, кортежи, множества»
Содержание

1. Цель и задачи лабораторной работы №2
2. Формулирование предметной области задачи
3. Знакомство со структурами данных в Python
4. Создание проекта программного приложения
5. Написание программного кода по заданному алгоритму
6. Запуск и тестирование результатов работы программы

Контрольные вопросы

1. Назовите два способа создания списка, содержащего пять целочисленных значений, равных нулю.
2. Назовите четыре операции, которые изменяют непосредственно объект словаря.
3. Как определить размер кортежа? Почему этот инструмент стоит обособленно?

Лабораторная работа 3. «Файлы»
Содержание:

1. Цель и задачи лабораторной работы №2
2. Формулирование предметной области задачи
3. Знакомство с файлами в Python
4. Создание проекта программного приложения
5. Написание программного кода по заданному алгоритму
6. Запуск и тестирование результатов работы программы

Контрольные вопросы

1. Какое значение используется по умолчанию в аргументе режима обработки файла в функции *open*?
2. Каким модулем можно воспользоваться для сохранения объектов в файл, чтобы избежать выполнения преобразований объектов в строки вручную?
3. Какие базовые режимы обработки файла существуют в Python?

Лабораторная работа 4. «Работа с операционной системой»

Содержание:

1. Цель и задачи лабораторной работы №4
2. Формулирование предметной области задачи
3. Знакомство с модулями для работы с ОС в Python
4. Создание проекта программного приложения
5. Написание программного кода по заданному алгоритму
6. Запуск и тестирование результатов работы программы

Контрольные вопросы

1. Как работает функция перемещения в модуле *shutil*?
2. Можно ли удалить файл с помощью модуля *os?*
3. Какие параметры нужны для подключения к удаленной машине по ssh?

Семестр 5

Лабораторная работа 5. «Общие принципы ООП»
Содержание:

1. Цель и задачи лабораторной работы №5
2. Формулирование предметной области задачи
3. Знакомство с ООП в Python
4. Создание проекта программного приложения
5. Написание программного кода по заданному алгоритму
6. Запуск и тестирование результатов работы программы

Контрольные вопросы

1. Каково основное назначение ООП в языке Python?
2. Как создать класс?
3. Какие две концепции ООП являются наиболее важными в языке Python?

Лабораторная работа 6. «Графический интерфейс на Python»
Содержание:

1. Цель и задачи лабораторной работы №6
2. Формулирование предметной области задачи
3. Знакомство с построением графического интерфейса в Python
4. Создание проекта программного приложения
5. Написание программного кода по заданному алгоритму
6. Запуск и тестирование результатов работы программы

Контрольные вопросы

1. Перечислите основные графические компоненты модуля *tkinter*
2. Какие виды менеджеров разметки есть в модуле *tkinter*? Как они размещают элементы?
3. Как загрузить звуки в программу с помощью модуля winsound?

3.2.2 Курсовой проект

5 семестр

Целью курсового проектирования является систематизация, закрепление и расширение теоретических знаний, приобретение необходимых практических навыков в области профессиональных задач, связанных с проектированием и разработкой программных компонентов операционных систем и системных утилит как в виде цельных комплексов, так и отдельных модулей.

Курсовой проект состоит из двух частей. Первая часть предполагает выполнение описания и проведение анализа предметной области и разработка концептуальной модели предметной области, выбранной обучающимся из предложенного списка. Во второй части предполагается составление и описание алгоритма и структуры программы, описание используемых модулей/библиотек. Третья часть предполагает реализацию программного приложения на языке высокого уровня Python.

Расчетно-пояснительная записка включает введение, три основных раздела (анализа требований, проектного, реализации), заключение и список использованной литературы.

При выполнении курсового проекта обучающиеся осваивают общие компетенции, такие как ОК2, ОК3, ОК4, ОК8, ОК9, и все профессиональные компетенции ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6

Возможный перечень тем для курсового проекта:

1. Разработка простого калькулятора на Python
2. Разработка игры в города на Python
3. Разработка экспертной системы оценки стоимости ремонта после ДТП на Python
4. Разработка игры "Кто хочет стать миллионером" на Python
5. Разработка модели чат-бота на Python
6. Разработка игры "Дуэль рыцарей" на Python
7. Разработка конвертера валют на Python
8. Разработка утилиты для поиска файла по названию
9. Разработка конвертера мер и весов на Python
10. Разработка программы-теста по теме "Системное программирование" на Python
11. Разработка игры "Дуэль ковбоев" на Python
12. Разработка системы хранения истории болезней пациентов на Python
13. Разработка системы управления заказами в кафе на Python
14. Разработка простого текстового квеста на Python
15. Разработка игры "Угадай число" на Python
16. Разработка программы шифрования методом Цезаря на Python
17. Разработка генератора имен индейцев на Python
18. Разработка системы цензурирования слов в тексте на Python
19. Разработка системы управления турами туристического агенства на Python
20. Разработка модуля поиска совпадения слов и фраз в комментариях на Python
21. Разработка генератора имен предметов Diablo 3 на Python
22. Разработка игры "Скачки" на Python
23. Разработка функции хэширования строки на Python
24. Разработка игры "Бросок мяча в кольцо" на Python
25. Разработка системы управления автопарком такси на Python
26. Разработка русско-английского переводчика на Python
27. Разработка модуля музыкального плейлиста на Python
28. Разработка архиватора файлов в zip-архив на Python
29. Разработка системы управления списком покупок на Python
30. Разработка игры "Крестики-нолики" на Python
31. Разработка функции генератора случайных строк на Python
32. Разработка утилиты для массового удаления файлов на Python
33. Разработка парсера текста на Python

3.2.3 Промежуточная аттестация

Зачетные билеты, 4 семестр

Билет 1

1. Синтаксис описания функции.
2. Способы передачи данных в функцию

Билет 2

1. Описание и свойства списка
2. Способы инициализации списка

Билет 3

1. Разница между title() и upper()
2. Свойства команды считывания строки с клавиатуры

Билет 4

1. Допустим, что имеется строка S со значением “s,pa,m”. Укажите два способа извлечения двух символов в середине строки.
2. Расскажите, как можно преобразовать символы в их ASCII-коды.

Билет 5

1. Опишите, для чего служит метод словаря popitem()
2. Расскажите значение слов True и False

Билет 6

1. Опишите различия между while и for
2. Описание и свойства кортежа

Билет 7

1. Назовите основные режимы доступа к файлу
2. Назовите три способа, с помощью которых можно присвоить одно и то значение трем переменным.

Билет 8

1. Опишите преимущества использования функций
2. Опишите принцип работы итераторов

Билет 9

1. Расскажите, как можно задать значение по умолчанию в функции
2. Опишите назначение функции range() в цикле for

Билет 10

1. Опишите возможные способы получения значения из списка (минимум два)
2. Расскажите, в каком случае точка с запятой в конце строки необходима

3.2.3 Учебная практика

4 семестр

Виды работ:

- проведение работ по удаленной настройке сервера с помощью ssh
- разработка программ для создания бэкапов
- настройка cron для периодического выполнения задач
- передача данных между устройствами с помощью сетевых интерфейсов
- разработка программ «родительского» контроля

3.2.4 Производственная практика

8 семестр

Виды работ:

- анализ предметной области;
- разработка концептуальной модели предметной области на основе анализа предметной области посредством построения функциональных моделей «Как есть» и «Как надо»;
- описание алгоритма работы программы;
- определение требуемых для работы модулей/библиотек;
- построение логической модели данных в среде моделирования Open Model Sphere и ее документирование;
- разработка программного приложения;
- тестирование программы;
- составление технической документации и оформление руководства пользователя

3.2.5 Квалификационный экзамен

Список вопросов теста

1. PYTHON является:
    a. Машинно - ориентированным языком (низкого уровня)
    b. Языком высокого уровня
    c. Объектно - ориентированным языком
2. Область применения PYTHON:
    a. Робототехника и искусственный интеллект
    b. Обучение
    c. Интернет
3. Год разработки PYTHON:
    a. 1990
    b. 1991
    c. 1993
4. Чувствителен ли PYTHON к регистру (большая или маленькая буквы):
    a. Да
    b. Нет
5. Какие существуют типы переменных (выбрать несколько выриантов):
    a. float
    b. list
    c. num
    d. int
    e. bool
    f. integer
6. Переменная int:
    a. вещественная переменная
    b. символьная строка
    c. логическая переменная
    d. целая переменная
7. Переменная str:
    a. символьная строка
    b. логическая переменная
    c. целая переменная
8. Переменная float:
    a. целая переменная
    b. вещественная переменная
    c. логическая переменная
9. Каков будет результат выполнения int("88"):
    a. "88"
    b. 88
    c. 88.00
10. Каков будет результат выполнения str(88):
    a. "88"
    b. 88
    c. 88.00
11. Имена переменных не могут включать:
    a. Русские буквы
    b. Латинские буквы
    c. Пробелы
    d. Скобки, знаки + = ! ? и др.
12. Какие имена являются правильными в PYTHON (выбрать несколько):
    a. N

b. ABC
c. sum
d. 41And
e. A+B
f. _mam

13. Что будет в результате выполнения команды:

```
a = 20
b = a + 5
a = b * 100
print(a)
```

a. 25
b. 2500
c. 25000
d. 1000

14. Что будет в результате следующего действия *print(2**20)*
a. 104576
b. 1048576
c. 964
d. 2
15. Что будет в результате выполнения следующего действия *print(23 % 2)*
a. 11
b. 1
c. 0
16. Результатом вычисления *print(24 // 3)* будет число:
a. 4
b. 8
c. 12
17. Что будет результатом выполнения алгоритма:
Входные данные: a = 5, b = 7

```
a = int(input())
b = int(input())
s = a + b
print(s)
```

a. 57
b. 12
c. 35

18. Что будет результатом выполнения алгоритма:
Входные данные: a = 5, b = 7

```
a = input()
b = input()
s = a + b
print(s)
```

a. 12
b. 57
c. 35

19. Что будет в результате выполнения следующего алгоритма:
Входные данные: -57

```
x = int(input())
if x > 0:
   print(x)
```

```
else:
   print(-x)
```

a. -57
b. 57
c. 0
d. -1

20. Что будет в результате выполнения программы:
Входные данные: a = 10, b = 20

```
a = int(input())
b = int(input())
if a < b:
   print(a)
else:
   print(b)
```

a. 10
b. 20
c. 30
d. -10

21. Какой ряд чисел может образоваться после выполнения следующего алгоритма:

```
for i in range(1,10):
      print(i)
```

a. 1 2 3 4 5 6 7 8 9 10
b. 1 2 3 4 5 6 7 8 9
c. 0

22. Какой ряд чисел может образоваться после выполнения алгоритма:

```
for i in range(1,10+1):
print(i)
```

a. 1 2 3 4 5 6 7 8 9 10
b. 1 2 3 4 5 6 7 8 9 10 11
c. 1 4 9 16

23. Что выведет программа после выполнения данного алгоритма:
Входные данные: name = Иванов

```
print('Как Ваша фамилия?')
name = input()
print('Здравствуйте, '+ name + '!' )
```

a. Как Ваша фамилия? Здравствуйте, Иванов!
b. Как Ваша фамилия? Здравствуйте, Иванов !
c. Как Ваша фамилия? Здравствуйте, Иванов !

24. Как обозначается логический оператор И, ИЛИ, НЕ в питоне:
a. OR, NOT, IF
b. AND, OR, NOT
c. AND, OR, IF
d. AND, ELSE, NOT

25. Что будет в результате выполнения следующего алгоритма программы:
Входные данные: a = 15, b = 45

```
a = int(input())
b = int(input())
if a % 10 == 0 or b % 10 == 0:
   print('YES')
else:
   print('NO')
```

a. YES

b. NO

26. Как будет записано число 18 после выполнения следующего алгоритма:

```
x = float(input())
print(x)
```

a. 18
b. 18.0
c. 18.00

27. Что будет после выполнения следующего листинга программы:

```
for i in range(4)
    print(i)
    print(i ** 2)
```

a. 0 0 1 1 3 3 4 4
b. 0 0 1 1 2 4 3 4
c. 0 0 1 1 2 3 3 9

28. Результатом выполнения алгоритма цикла while будет:

```
i = 1
while i <= 10:
    print(i ** 2)
    i = i + 1
```

a. 1 2 4 8 12 14
b. 1 2 16 24 32
c. 1 2 4 16 25 36 49 64 81 100

4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИИ

4.1. Система оценивания компетенций

Система оценивания компетенций, формируемых в ходе освоения дисциплины, основана на оценивании всех видов работ.

В таблицах 4.1.1-4.1.3 – система оценивания каждого вида работ.

Таблица 4.1.1 – Критерии оценивания лабораторной работы:

Уровень усвоения	Описание
Отлично	Правильное выполнение задания и точные ответы на все контрольные и дополнительные вопросы. Сформированные умения по применению полученных знаний в решении профессиональных задач в рамках учебного материала.
Хорошо	Выполнение задания с небольшими неточностями и правильные ответы на большинство контрольных и дополнительных вопросов. В целом успешные, но содержащие отдельные пробелы умения по применению полученных знаний в решении профессиональных задач в рамках усвоенного учебного материала.
Удовлетворительно	Выполнение задания с существенными неточностями и ответы с затруднениями на большинство контрольных и дополнительных вопросов. В целом успешные, но не систематически осуществляемые умения по применению полученных знаний в решении профессиональных задач в рамках усвоенного учебного материала.
Неудовлетворительно	Неправильное выполнение задания, множество неточностей в ответах на контрольные и дополнительные вопросы.

	Частично освоенные или полностью неосвоенные умения по применению полученных знаний в решении профессиональных задач в рамках учебного материала.

Таблица 4.1.2 – Критерии оценивания курсового проекта

Уровень усвоения	Описание
Отлично	Курсовой проект выполнен в полном объеме и соответствует заданию. Пояснительная записка составлена аккуратно, последовательно с учетом требований стандартов по составлению текстовых документов; практическая часть курсового проекта выполнена в полном объеме: база данных нормализована и соответствует предметной области, программное приложение – надёжно и эффективно, имеет удобный пользовательский интерфейс; выполнение курсового проекта проходило в полном соответствии с графиком курсового проектирования; защита курсового проекта проведена грамотно с демонстрацией всех возможностей разработанного программного приложения и базы данных.
Хорошо	Курсовой проект выполнен в полном объеме и соответствует заданию. Пояснительная записка составлена аккуратно, но имеются незначительные погрешности в ее оформлении. практическая часть курсового проекта выполнена в полном объеме, но имеет несущественные замечания: база данных нормализована и соответствует предметной области, имеются небольшие замечания к интерфейсу программного приложения и его устойчивости; имеются некоторые отступления от графика выполнения курсового проектирования.
Удовлетворительно	Курсовой проект имеет ошибки и неточности, непоследовательность в оформлении пояснительной записки. Имеются значительные отступления от требований ЕСКД при выполнении графической части проекта; отсутствие самостоятельности и творческого подхода при разработке программного приложения; существование незначительных погрешностей в работе программного приложения; значительное отступление от сроков выполнения курсового проекта; недостаточно грамотная защита и неполная демонстрация возможностей разработанного программного приложения.
Неудовлетворительно	Курсовой проект не соответствует заданию; отсутствует учет требований стандартов по оформлению текстовых документов при оформлении пояснительной записки; отсутствует учет требований стандартов ЕСКД при выполнении графической части проекта; имеются ошибки и непоследовательность в работе программного приложения; база данных не соответствует требованиям проектирования; значительное отступление от сроков выполнения курсового проекта; неспособность грамотно защитить курсовой проект.

Таблица 4.1.3 – Критерии оценки: диф. зачета 5 семестр

Уровень усвоения	Описание
Отлично	Глубокие исчерпывающие знания всего программного материала, последовательные, содержательные, полные, правильные и конкретные ответы на оба вопроса экзаменационного билета и дополнительные вопросы преподавателя.
Хорошо	Твердые и достаточно полные знания всего программного материала, последовательные, правильные, конкретные ответы на поставленные вопросы при свободном устранении замечаний по отдельным вопросам.
Удовлетворительно	Знание и понимание основного программного материала; правильные, без грубых ошибок ответы на поставленные вопросы при устранении неточностей и несущественных ошибок в освещении отдельных положений при наводящих вопросах преподавателя.
Неудовлетворительно	Грубые ошибки в ответе, непонимание сущности излагаемых вопросов; неуверенные и неточные ответы на дополнительные вопросы.

4.2. Итоговая оценка по дисциплине

Оценка уровня усвоения компетенций производится на основе средней оценки по всем видам работ: лабораторных работ, курсового проекта и дифференцированного зачета.

Оценки ставятся по 5-балльной шкале. Округление оценки производится в пользу студента. Итоговая оценка выставляется в ведомость согласно следующему правилу:

Итоговая оценка по дисциплине			
Неудовлетворительно	**Удовлетворительно**	**Хорошо**	**Отлично 5**
0 – 2.9	3.0-3.49	3.5-4.4	4.5-5.0

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Восточно-Сибирский государственный университет технологий управления»

Технологический колледж
Кафедра «Системы информатики»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

к дисциплине МДК.01.02 «Прикладное программирование»

Специальность: 09.02.03 - Программирование в компьютерных системах

Форма обучения: Очная

Присваиваемая квалификация: Техник-программист

Улан-Удэ
2018

Паспорт
фонда оценочных средств
по дисциплине МДК.01.02 «Прикладное программирование»

1. ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

В результате освоения дисциплины у обучающихся должны быть сформированы следующие компетенции:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии в профессиональной деятельности.

ОК 6. Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

ПК 1.1 Выполнять разработку спецификаций отдельных компонент.

ПК 1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4 Выполнять тестирование программных модулей.

ПК 1.5 Осуществлять оптимизацию программного кода модуля.

ПК 1.6 Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.

2. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ НА РАЗЛИЧНЫХ ЭТАПАХ ИХ ФОРМИРОВАНИЯ, ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ

2.1. Описание показателей оценивания компетенций

Фонд оценочных средств сформирован на основе ключевых принципов оценивания:

- валидность: объекты оценки должны соответствовать поставленным целям обучения;
- надежность: использование единообразных показателей и критериев для оценивания достижений;
- объективность: получение объективных и достоверных результатов при проведении контроля с различными целями.

В процессе оценивания знаний, умений и навыков обучающийся должен демонстрировать следующие результаты по освоению компетенций:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

Знать: как и где можно применить теоретические знания и практические умения, полученные в результате освоения дисциплины в различных сферах деятельности.

ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

Уметь: применять теоретические знания и практические умения для выполнения индивидуальных заданий, курсовых проектов, заданий учебной и производственной практик и выпускной квалификационной работы.

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

Уметь: самостоятельно анализировать предметную область при выборе объекта автоматизации в период прохождения производственной практики.

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

Уметь: находить информацию, дающую исчерпывающее представление о той предметной области, для которой разрабывается база данных и программное приложение в рамках курсового проектирования или выпускной квалификационной работы

ОК 5. Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.

Уметь: выполнять форматирование и оформление текстовых, графических документов, расчеты в электронных таблицах и представление числовой информации в виде диаграмм и графиков для анализа данных.

ОК 6. Работать в коллективе и команде, обеспечивать ее сплочение, эффективно общаться с коллегами, руководством, потребителями.

Уметь: строить рабочие и коммуникативные отношения в период учебной и производственной практик.

ОК 7. Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий.

Уметь: строить рабочие и коммуникативные отношения в малых группах при выполнении лабораторных работ, решения ситуационных задач на интерактивных занятиях.

ОК 8. Самостоятельно определять задачи, заниматься самообразованием, осознанно планировать повышение квалификации.

Уметь: оценивать собственный уровень профессионального и личностного развития, определять пути самообразования.

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

Уметь: определять новые технологии и быть готовым к их освоению и использованию в профессиональной деятельности.

ПК 1.1 Выполнять разработку спецификаций отдельных компонент.

Иметь практический опыт: разработки спецификаций программных компонент;

Знать: основные принципы разработки компонент программных средств; основные принципы построения спецификации отдельных компонент; современные методы и инструментальные средства разработки спецификации;

Уметь: разрабатывать спецификации отдельных компонент программных средств;

ПК 1.2 Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

Иметь практический опыт: создания программ по разработанным алгоритмам для решения поставленных задач;

Знать: синтаксис и особенности конкретного языка программирования, на котором будет реализован модуль; основы алгоритмизации процессов;

Уметь: создавать программу по разработанному алгоритму как отдельный модуль;

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.

Иметь практический опыт: работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок;

Знать: основные принципы отладки и тестирования программных продуктов;

Уметь: выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся;

ПК 1.4 Выполнять тестирование программных модулей.

Иметь практический опыт: проведения тестирования программы;

Знать: существующие методы тестирования программ; основные принципы проведения тестирования

Уметь: проводить тестирование программного модуля по определенному алгоритму;

ПК 1.5 Осуществлять оптимизацию программного кода модуля.

Иметь практический опыт: проведения оптимизации программы для повышения эффективности и читабельности кода;

Знать: методы оптимизации программного кода; основные способы сокращения объема кода программы и улучшения читабельности кода;

Уметь: использовать программные инструменты для уменьшения объема кода программы; проводить проверку программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы.

ПК 1.6 Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.

Иметь практический опыт: разработки документации программных средств и описания алгоритмов решения поставленных задач;

Знать: методы и средства разработки проектной и технической документации для описания программных средств и алгоритмов решения задач;

Уметь: создавать проектную и техническую документацию к программным средствам; описывать алгоритмы решения поставленных задач.

Описание показателей, форм и методов оценивания компетенций представлены в таблице 2.1, в которой отображены процессы формирования вклада дисциплины «Технология разработки и защиты баз данных» в формирование компетенций с ПК2.1 по ПК2.4 и с ОК-1 по ОК-9.

Таблица 2.1. Показатели оценивания компетенций на различных этапах их формирования

№ п/п	Контролируемые разделы (темы) дисциплины*	Код контролируемой компетенции **	Наименование оценочного средства
1	**Раздел 1.** Основы программирования	ОК 1, ОК 2, ОК 6, ПК 1.1, ПК 1.6	Вопросы по лабораторной работе
2	**Раздел 2.** Основы структурного программирования	ОК 1, ОК 3 – 8, ПК 1.1 - 1.6	Зачетные билеты Экзаменационные билеты Вопросы по лабораторной работе
3	**Раздел 3.** Программирование с учетом внешних факторов	ОК 3, ОК 6, ОК 7, ОК 9, ПК 1.2 - 1.5	Экзаменационные билеты Вопросы по лабораторной работе
4	**Раздел 4.** Объектно-ориентированное программирование	ОК 3 - 5, ОК 7 – 9, ПК 1.1 - ПК 1.5	Экзаменационные билеты Вопросы по лабораторной работе

2.2. Критерии оценивания компетенций, описание шкал оценивания

Описание критериев и шкал оценивания компетенций ПК-1.1, ПК-1.2, ПК-1.3, ПК-1.4, ПК-1.5, ПК-1.6 в рамках оценивания результатов обучения по профессиональному модулю «Разработка

программных модулей программного обеспечения для компьютерных систем» приведено в таблицах 2.2.1, 2.2.2, 2.2.3, 2.2.4, 2.2.5, 2.2.6 соответственно.

Таблица 2.2.1 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.1

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание основных принципов разработки компонент программных средств; основных принципов построения спецификации отдельных компонент; современных методов и инструментальные средства разработки спецификации. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание основных принципов разработки компонент программных средств; основных принципов построения спецификации отдельных компонент; современных методов и инструментальные средства разработки спецификации.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание основных принципов разработки компонент программных средств; основных принципов построения спецификации отдельных компонент; современных методов и инструментальные средства разработки спецификации.	Сформированные систематические знания, глубокое понимание основных принципов разработки компонент программных средств; основных принципов построения спецификации отдельных компонент; современных методов и инструментальные средства разработки спецификации.
	Отсутствие сформированных умений / частично освоенные умения разрабатывать спецификации отдельных компонент программных средств.	В основном сформированные умения разрабатывать спецификации отдельных компонент программных средств. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения разрабатывать спецификации отдельных компонент программных средств. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения разрабатывать спецификации отдельных компонент программных средств.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта разработки спецификаций программных компонент	В основном сформированные навыки при приобретении практического опыта разработки спецификаций программных компонент. В целом успешные навыки работы, но применяемые не систематически	Сформированные навыки при приобретении практического опыта разработки спецификаций программных компонент. В целом успешное применение опыта работы, но содержащее отдельные пробелы	Полностью приобретенный опыт разработки спецификаций программных компонент. Успешное и систематическое применение навыков работы

Таблица 2.2.2 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.2

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание синтаксиса и особенностей конкретного языка программирования, на котором будет реализован модуль; основ алгоритмизации процессов. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание синтаксиса и особенностей конкретного языка программирования, на котором будет реализован модуль; основ алгоритмизации процессов.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание синтаксиса и особенностей конкретного языка программирования, на котором будет реализован модуль; основ алгоритмизации процессов.	Сформированные систематические знания, глубокое понимание синтаксиса и особенностей конкретного языка программирования, на котором будет реализован модуль; основ алгоритмизации процессов.
	Отсутствие сформированных умений / частично освоенные умения создавать программу по разработанному алгоритму как отдельный модуль.	В основном сформированные умения создавать программу по разработанному алгоритму как отдельный модуль. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения создавать программу по разработанному алгоритму как отдельный модуль. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения создавать программу по разработанному алгоритму как отдельный модуль.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта создания программ по разработанным алгоритмам для решения поставленных задач.	В основном сформированные навыки при приобретении практического опыта создания программ по разработанным алгоритмам для решения поставленных задач. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта создания программ по разработанным алгоритмам для решения поставленных задач. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт создания программ по разработанным алгоритмам для решения поставленных задач. Успешное и систематическое применение навыков работы.

Таблица 2.2.3 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.3

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично

Критерии	Фрагментарные знания и понимание основных принципов отладки и тестирования программных продуктов. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание основных принципов отладки и тестирования программных продуктов.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание основных принципов отладки и тестирования программных продуктов.	Сформированные систематические знания, глубокое понимание основных принципов отладки и тестирования программных продуктов.
	Отсутствие сформированных умений / частично освоенные умения выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся.	В основном сформированные умения выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения выполнять отладку, используя встроенные в среду разработки средства отладки программ; проводить разбор кода, написанного самостоятельно либо другим обучающимся.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок.	В основном сформированные навыки при приобретении практического опыта работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт работы со средствами отладки программ, разбора кода для тестирования программы или поиска ошибок. Успешное и систематическое применение навыков работы.

Таблица 2.2.4 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.4

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание существующих методов тестирования программ; основных принципов	Общие, но не структурированные знания и понимание существующих методов тестирования	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание	Сформированные систематические знания, глубокое понимание существующих

	проведения тестирования. Отсутствие знаний и понимания их отличительных особенностей.	программ; основных принципов проведения тестирования.	существующих методов тестирования программ; основных принципов проведения тестирования.	методов тестирования программ; основных принципов проведения тестирования.
	Отсутствие сформированных умений / частично освоенные умения проведения тестирование программного модуля по определенному алгоритму.	В основном сформированные умения проведения тестирование программного модуля по определенному алгоритму. В целом успешные умения, но осуществляемые не систематически.	Сформированные умения проведения тестирование программного модуля по определенному алгоритму. В целом успешные, но содержащие отдельные пробелы.	Полностью сформированные умения проведения тестирование программного модуля по определенному алгоритму.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта проведения тестирования программы.	В основном сформированные навыки при приобретении практического опыта проведения тестирования программы. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта проведения тестирования программы. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт проведения тестирования программы. Успешное и систематическое применение навыков работы.

Таблица 2.2.5 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.5

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание методов оптимизации программного кода; основных способов сокращения объема кода программы и улучшения читабельности кода. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание методов оптимизации программного кода; основных способов сокращения объема кода программы и улучшения читабельности кода.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание методов оптимизации программного кода; основных способов сокращения объема кода программы и улучшения читабельности кода.	Сформированные систематические знания, глубокое понимание методов оптимизации программного кода; основных способов сокращения объема кода программы и улучшения читабельности кода.
	Отсутствие сформированных умений / частично освоенные умения использования	В основном сформированные умения использования программные инструменты для	Сформированные умения использования программные инструменты для уменьшения объема	Полностью сформированные умения использования программные инструменты для

	программные инструменты для уменьшения объема кода программы; проведения проверки программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы.	уменьшения объема кода программы; проведения проверки программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы. В целом успешные умения, но осуществляемые не систематически.	кода программы; проведения проверки программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы. В целом успешные, но содержащие отдельные пробелы.	уменьшения объема кода программы; проведения проверки программного кода с целью оптимизирования алгоритмов и ускорения выполнения программы.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта проведения оптимизации программы для повышения эффективности и читабельности кода.	В основном сформированные навыки при приобретении практического опыта проведения оптимизации программы для повышения эффективности и читабельности кода. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта проведения оптимизации программы для повышения эффективности и читабельности кода. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт проведения оптимизации программы для повышения эффективности и читабельности кода. Успешное и систематическое применение навыков работы.

Таблица 2.2.6 - Критерии оценивания результатов обучения по дисциплине «Прикладное программирование» как части дескрипторов компетенции ПК-1.6

Шкалы	Не освоена	Освоена частично	Освоена в основном	Освоена полностью
	Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
Критерии	Фрагментарные знания и понимание методов и средств разработки проектной и технической документации для описания программных средств и алгоритмов решения задач. Отсутствие знаний и понимания их отличительных особенностей.	Общие, но не структурированные знания и понимание методов и средств разработки проектной и технической документации для описания программных средств и алгоритмов решения задач.	Сформированные, но содержащие отдельные пробелы в знаниях и достаточно глубокое понимание методов и средств разработки проектной и технической документации для описания программных средств и алгоритмов решения задач.	Сформированные систематические знания, глубокое понимание методов и средств разработки проектной и технической документации для описания программных средств и алгоритмов решения задач.
	Отсутствие сформированных умений / частично освоенные умения создавать проектную и техническую	В основном сформированные умения создавать проектную и техническую документацию к	Сформированные умения создавать проектную и техническую документацию к программным	Полностью сформированные умения создавать проектную и техническую документацию к

	документацию к программным средствам; описывать алгоритмы решения поставленных задач.	программным средствам; описывать алгоритмы решения поставленных задач. В целом успешные умения, но осуществляемые не систематически.	средствам; описывать алгоритмы решения поставленных задач. В целом успешные, но содержащие отдельные пробелы.	программным средствам; описывать алгоритмы решения поставленных задач.
	Отсутствие сформированных навыков / частично сформированные навыки при приобретении практического опыта разработки документации программных средств и описания алгоритмов решения поставленных задач.	В основном сформированные навыки при приобретении практического опыта разработки документации программных средств и описания алгоритмов решения поставленных задач. В целом успешные навыки работы, но применяемые не систематически.	Сформированные навыки при приобретении практического опыта разработки документации программных средств и описания алгоритмов решения поставленных задач. В целом успешное применение опыта работы, но содержащее отдельные пробелы.	Полностью приобретенный опыт разработки документации программных средств и описания алгоритмов решения поставленных задач. Успешное и систематическое применение навыков работы.

3. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ И МАТЕРИАЛЫ, НЕОБХОДИМЫЕ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ И НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИИ

3.1 Типовые контрольные задания и материалы по дисциплине МДК.01.02 «Прикладное программирование»

3.1.1 Лабораторные работы

Семестр 5

Лабораторная работа 1. «Базовые типы данных, функции ввода-вывода на консоль»
Содержание:

1. Краткие теоретические сведения
   1.1 Создание проекта
   1.2 Порядок отладки программы
2. Постановка задачи
3. Пример решения индивидуального задания
4. Варианты индивидуальных заданий

Лабораторная работа 2. «Разветвление и условные операторы»
Содержание:

1. Краткие теоретические сведения
   1.1 Базовые типы данных
   1.2 Краткие сведения о способах ввода вывода
   1.3 Принципы структурного программирования
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 3. «Циклы»
Содержание:
1. Краткие теоретические сведения
   1.1 Операторы циклов
   1.2 Операторы перехода
   1.3 Принципы структурного программирования
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 4. «Массивы»
Содержание:
1. Краткие теоретические сведения
   1.1 Одномерные массивы
   1.2 Использование многомерных массивов
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 5. «Символьные строки»
Содержание:
1. Краткие теоретические сведения
   1.1 Массив символов и его инициализация
   1.2 Функции для обработки строк
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 6. «Функции»
Содержание:
1. Краткие теоретические сведения
   1.1 Объявление функций
   1.2 Аргументы функции
   1.3 Функции, возвращающие значение
   1.4 Функции, не возвращающие значение
   1.5 Передача массива в функции
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 7. «Указатели»
Содержание:
1. Краткие теоретические сведения
   1.1 Указатели
   1.2 Указатели и массивы
   1.3 Динамическое распределение памяти
   1.4 Пример динамического выделения памяти для массивов
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 8. «Пользовательские типы данных. Структуры»
Содержание:
1. Краткие теоретические сведения
   1.1 Структуры
   1.2 Объявление шаблонов структур

1.3 Определение структур-переменных
1.4 Доступ к компонентам (элементам) структуры
1.5 Инициализация структур
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 9. «Динамические списки»
Содержание:
1. Краткие теоретические сведения
1.1 Механизмы доступа
1.2 Связанный список
1.3 Пример организации связанного списка
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 10. «Файловый ввод-вывод»
Содержание:
1. Краткие теоретические сведения
1.1 Потоки и файлы
1.2 Работа с файлами на основе конструкций языка С
1.3 Работа с файлами на основе потоков языка С++
2. Постановка задачи
3. Варианты индивидуальных заданий

Семестр 6
Лабораторная работа 1. «Программирование с учетом внешних факторов»
Содержание:
1. Краткие теоретические сведения
1.1 Исключения и ошибки
1.2 Директивы препроцессора
1.3 Оконное приложение
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 2. «Класс»
Содержание:
1. Краткие теоретические сведения
1.1 Понятие класса
1.2 Пример класса
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 3. «Конструкторы и деструкторы»
Содержание:
1. Краткие теоретические сведения
1.1 Конструкторы
1.2 Деструкторы
1.3 Статические элементы класса
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 4. «Права доступа внутри класса»
Содержание:
1. Краткие теоретические сведения
   1.1 Права доступа к компонентам класса
   1.2 Дружественные функции и классы
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 5. «Перегрузка операций»
Содержание:
1. Краткие теоретические сведения
   1.1 Способы перегрузки операции
   1.2 Правила перегрузки операторов
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 6. «Наследование»
Содержание:
1. Краткие теоретические сведения
   1.1 Наследования
   1.2 Права доступа при наследовании
   1.3 Множественное наследование
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 7. «Абстрактные классы»
Содержание:
1. Краткие теоретические сведения
   1.1 Полиморфизм
   1.2 Виртуальные функции
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 8. «Шаблоны классов»
Содержание:
1. Краткие теоретические сведения
   1.1 Шаблоны
   1.2 Шаблон функции
   1.3 Шаблон класса
2. Постановка задачи
3. Варианты индивидуальных заданий

Лабораторная работа 9. «Потоковые и контейнерные классы»
Содержание:
4. Краткие теоретические сведения
   4.1 Потоковые классы
   4.2 Контейнерные классы
5. Постановка задачи
6. Варианты индивидуальных заданий

Лабораторная работа 10. «Итераторы и алгоритмы»
Содержание:

7. Краткие теоретические сведения
   7.1 Итераторы STL
   7.2 Функциональные объекты STL
   7.3 Библиотека algorithm
8. Постановка задачи
9. Варианты индивидуальных заданий

3.1.2 Промежуточная аттестация

Зачетные билеты, семестр 5

Билет 1.

1. Понятие алгоритма, свойства алгоритмов. Основные алгоритмические конструкции.

Формы записей алгоритмов.

2. Напишите программу, которая выполняет следующее:
   - считывает из файла «cars.txt» в массив структур;
   - выводит на экран в виде таблицы данные элементов массива;
   - добавляет в массив элемент вводом с клавиатуры;
   - записывает обратно в файл «cars.txt» измененный массив структур.

   Структура «Автомобиль» (Car). Поля: марка (name), год выпуска (date), цена (price).

Билет 2.

1. Структура программы на языке С. Операторы условной передачи управления. Операторы организации циклической обработки. Операторы перехода.
2. Напишите программу, которая выполняет следующее:
   - считывает из файла «info.txt» в массив элементы структуры;
   - выводит на экран количество элементов;
   - добавляет в массив 3 элемента вводом с клавиатуры;
   - записывает обратно в файл «info.txt» измененный массив структур.

   Структура «Информация» (Info). Поля: носитель (carrier), объем (volume), название (name).

Билет 3.

1. Определение понятия «строка». Объявление (инициализация) строки. Строковые константы. Функции для ввода и вывода строк.
2. Напишите программу, которая выполняет следующее:
   - создает динамический массив целых чисел;
   - считывает элементы массива с клавиатуры или из файла «input.txt» в зависимости от того, введет пользователь 1 или 2.

- выводит этот массив на экран или в файл «output.txt» в зависимости от того, введет пользователь 1 или 2;
- очищает память от массива.

Билет 4.

1. Определение понятия «функция». Форма объявления функции. Прототип функции. Аргументы функции. Возвращаемое значение функции. Процедура.
2. Напишите программу, которая выполняет следующее:
   - считывает из файла «cities.txt» в массив элементы структуры;
   - сортирует по названию и выводит на экран;
   - добавляет в массив элемент вводом с клавиатуры;
   - записывает в файл «cities.txt» измененный массив структур.

   Структура «Город» (City). Поля: название (name), страна (country), население (population).

Билет 5.

1. Определение понятия «указатель». Форма объявления указателя. Получение значения из указателя. Адресная арифметика.
2. Напишите программу, в которой используется динамический список целых чисел (в структуре будет только одно поле данных – целое число). В ней должны быть следующие функции:
   - добавить элемент в список (в начало);
   - вывести элементы на экран с конца (можно воспользоваться рекурсивной функцией).

Билет 6.

1. Определение понятия «указатель». Форма объявления указателя. Динамическое управление памятью. Функции malloc() и free() / команды new и delete.
2. Напишите программу, которая выполняет следующее:
   - считывает из файла «data.txt» массив вещественных чисел;
   - считывает с клавиатуры число (возможен ввод числа 0, нужно учесть это);
   - делит все элементы массива на заданное число (перезаписывает массив);
   - записывает измененный массив в файл «data.txt».

Билет 7.

1. Виды пользовательских типов данных. Структура. Объявление структуры. Элементы и экземпляры структуры. Размещение в памяти экземпляра структуры. Доступ к элементам структуры (также и по указателю). Оператор sizeof.
2. Напишите программу, которая выполняет следующее:
   - считывает массив вещественных чисел из файла «num.txt» в динамический массив;
   - вычисляет среднее значение массива и добавляет его как новый элемент массива;

- записывает массив обратно в файл «num.txt»;
- освобождает память от массива.

Должна быть отдельная функция для среднего значения и добавления его в массив.

Билет 8.

1. Виды пользовательских типов данных. Объединение. Объявление объединения. Элементы и переменные (экземпляры) объединения. Размещение в памяти экземпляра объединения. Доступ к элементам объединения. Оператор sizeof.
2. Напишите программу, в которой используется динамический список символов (в структуре будет только одно поле данных – символ). В ней должны быть следующие функции:
   - добавить элемент в конец списока;
   - вывести список на экран;
   - записать список в файл «output.txt»

Билет 9.

1. Виды пользовательских типов данных. Перечисление. Объявление перечисления. Что представляют из себя элементы перечисления? Операторы sizeof и typedef.
2. Напишите программу, в которой используется динамический список целых чисел (в структуре будет только одно поле данных – целое число). В ней должны быть следующие функции:
   - добавить элемент в начало списка;
   - прокрутить список – элемент из начала переносится в конец списка;
   - вывести список на экран;

Билет 10.

1. Механизм работы потоков и файлов. Стандартные потоки.
2. Напишите программу, которая выполняет следующее:
   - считывает из файла «films.txt» в массив элементы структуры;
   - выводит на экран в виде таблицы данные фильмов заданного жанра;
   - добавляет в массив элемент вводом с клавиатуры;
   - записывает в файл «films.txt» измененный массив структур.

   Структура «Фильм» (Film). Поля: название (name), режиссер (director), жанр (genre).

Билет 11.

1. Переменные типа FILE. Указатель файла. Открытие файла. Типы доступа. Макрос FOPEN_MAX**.** Закрытие файла. Макрос EOF.
2. Напишите программу, в которой используется динамический список строк (в структуре будет только одно поле данных – строка). В ней должны быть следующие функции:
   - добавить элемент в начало списка;

- удалить элемент с заданным номером из списка;
- вывести элементы на экран.

Билет 12.

1. Механизм доступа к данным. Организация списка данных с использованием структуры.
2. Напишите программу, которая выполняет следующее:
   - создает динамический массив чисел заданного размера;
   - удаляет из массива минимальный элемент (если их несколько, то все найденные);
   - выводит на экран массив;
   - освобождает память от массива после работы с ним.

   В программе должна быть отдельная функция для удаления минимального элемента, которая принимает в качестве аргументов массив и количество элементов, возвращает измененное количество элементов.

Экзаменационные билеты, семестр 6

Билет 1.

1. Переменные и константы. Базовые типы данных языка С и их модификация. Простые операторы: оператор присваивания, арифметические и логические операторы, операторы сравнения. Выражения: порядок вычислений, преобразование типов в выражениях.
2. Объектно-ориентированное программирование. Понятие класса. Его отличия от структуры. Пример класса.

Билет 2.

1. Массивы как структурированный тип данных. Расположение массива в памяти. Инициализация числовых массивов. Безразмерные массивы.
2. Конструкторы и деструкторы класса. Механизм их работы. Статические элементы класса. Чем отличаются от других элементов.

Билет 3.

1. Структурное программирование. Определение понятия «функция». Форма объявления функции. Область видимости функции и внутри функции.
2. Виды прав доступа внутри класса. Определение дружественных функций и классов.

Билет 4.

1. Определение понятия «функция». Форма объявления функции. Рекурсивные функции, как работает рекурсия.
2. Перегрузка функций. Перегрузка операторов. Правила и способы перегрузки операторов.

Билет 5.

1. Определение понятия «указатель». Форма объявления указателя. Определение понятия «ссылка». Форма объявления ссылки. Отличия ссылки от указателя.

2. Устойчивость программной системы. Механизм исключительных ситуаций. Обработка программных исключений. try, catch, throw. Блок finally.

Билет 6.

1. Определение понятия «указатель». Форма объявления указателя. Создание динамических одномерных и двумерных массивов.
2. Устойчивость программной системы. Классические подходы к обработке ошибок. Их недостатки. Определение понятия «исключение». Виды исключений.

Билет 7.

1. Виды пользовательских типов данных. Структура. Объявление структуры. Элементы и экземпляры структуры. Механизм присваивания структур. Способы передачи структуры в функцию. Оператор typedef.
2. Механизм наследования в объектно-ориентированном программировании. Права доступа при наследовании. Множественное наследование.

Билет 8.

1. Виды пользовательских типов данных. Битовые поля. Объявление битовых полей. Размещение в памяти битовых полей. Операторы typedef.
2. Механизм полиморфизма в объектно-ориентированном программировании. Виртуальные функции и абстрактные классы.

Билет 9.

1. Определения потока и файла. Отличия. Виды потоков.
2. Что называют директивами препроцессора. Подключение внешних файлов с кодом. Макроподстановка. Макросы с формальными параметрами. Директивы условной компиляции. Операторы препроцессора.

Билет 10.

1. Переменные типа FILE. Указатель файла. Функции для работы с файлами и потоками: fopen(), fclose(), fprintf(), fscanf(), feof(), fputs(), fgets(), fread(), fwrite(), rewind(), remove(), fflush().
2. Шаблоны в C++. Шаблон функции. Шаблон класса.

Билет 11.

1. Механизм доступа к данным. Виды механизмов доступа. Принципы работы очереди и стека.
2. Потоковые классы. Контейнерные классы.

Билет 12.

1. Поразрядные операции. Виды поразрядных операций. Операции сдвига. Типы данных, к которым применимы поразрядные операции.
2. Итераторы. Библиотека STL и функциональные объекты.

3.1.3 Учебная практика

6 семестр

Виды работ:

1) Решение задач по теме «Основы структурного программирования». Необходимо представить задания, требующие применения навыков работы с командами ветвления и циклов.
2) Решение задач по теме «Массивы и строки». Необходимо представить задания, требующие применения навыков работы с числовыми и символьными массивами с использованием функций по работе со строками.
3) Решение задач по теме «Функции». Необходимо представить задания, требующие применения навыков работы с функциями.
4) Решение задач по теме «Указатели и динамическое распределение памяти». Необходимо представить задания, требующие применения навыков работы с указателями и методами динамического распределения памяти.
5) Решение задач по теме «Пользовательские типы данных». Необходимо представить задания, требующие применения навыков работы со структурами, объединениями, перечислениями и битовыми полями.
6) Решение задач по теме «Сложные структуры данных». Необходимо представить задания, требующие применения навыков работы со связанными списками на основе структуры.
7) Решение задач по теме «Файловый ввод\вывод». Необходимо представить задания, требующие применения навыков работы с библиотеками <stdio.h> и <fstream> для обеспечения ввода\вывода в файл.
8) Решение задач по теме «Исключения и ошибки». Необходимо представить задания, требующие применения навыков работы с конструкциями, обеспечивающими программную обработку исключений.
9) Решение задач по теме «Классы. Конструкторы и деструкторы». Необходимо представить задания, требующие применения навыков работы с классами.
10) Решение задач по теме «Перегрузка операций». Необходимо представить задания, требующие применения навыков обеспечения перегрузки функций и операций.
11) Решение задач по теме «Наследование». Необходимо представить задания, требующие применения навыков построения иерархии классов и работы с особенностями свойств наследования.
12) Решение задач по теме «Абстрактные классы». Необходимо представить задания, требующие применения навыков создания абстрактных классов и виртуальных функций на базе иерархии классов.
13) Решение задач по теме «Шаблоны классов». Необходимо представить задания, требующие применения навыков построения шаблонов функций и классов.
14) Решение задач по теме «Потоковые и контейнерные классы». Необходимо представить задания, требующие применения навыков построения и контейнерных классов и применения специальных библиотек, содержащих такие классы.
15) Решение задач по теме «Итераторы и функциональные объекты». Необходимо представить задания, требующие применения навыков работы с библиотеками STL и algorithm.

3.1.4 Производственная практика

8 семестр

Необходимо разработать проект программного обеспечения и реализовать на его основе программное обеспечение, удовлетворяющее требованиям предметной области. Для этого необходимо выполнить следующие виды работ:

1) детальный анализ предметной области в соответствии с заданием. Документирование результатов анализа;
2) определение и уточнение требований и задач к проекту. Документирование этих требований;

3) проектирование алгоритмов решения задач в соответствии с требованиями. Документирование этих алгоритмов;
4) реализация алгоритмов как отдельных программных модулей. Необходимо использовать знания по структурному и объектно-ориентированному программированию, полученные ранее. Документирование полученных кодов программных модулей;
5) объединение программных модулей в единое программное обеспечение;
6) тестирование и документирование результатов тестирования программного обеспечения;
7) построение финального отчета о построении и реализации проекта программного обеспечения.

3.1.5 Квалификационный экзамен

Квалификационный экзамен проводится в форме тестирования.

Задания, оценивающие сформированную компетенцию ПК-1.1

1. Какую функцию должны содержать все программы на C++?
   a) main()
   b) program()
   c) system()
   d) func()
2. При уничтожении объекта класса автоматически вызывается специальный метод класса:
   a) Деструктор
   b) Конструктор
   c) Разрушитель
   d) delete
3. Сколько производных классов можно получить из базового класса?
   a) Количество производных классов определяется количеством базовых классов
   b) Неограниченное количество
   c) Только один
   d) Количество производных классов определяется количеством объектов базового класса
4. Какой служебный знак ставится после оператора case в конструкции switch?
   a) ;
   b) :
   c) .
   d) ->

Задания, оценивающие сформированную компетенцию ПК-1.2

5. Как называется принцип создания класса на базе существующего:
   a) Полиморфизм
   b) Наследование
   c) Инкапсуляция
   d) Метаморфизм
6. При объявлении структуры необходимо использовать следующее ключевое слово:
   a) structure
   b) struct
   c) object
   d) class
7. Типы данных есть в C\C++:
   a) int, real, char, for

b) float, int, if, string
c) int, float, char, bool
d) double, class, vector, new

8. Какому понятию подходит определение: объединяет в себе поля, методы и свойства:
   a) Переменная
   b) Перечисление
   c) Класс
   d) Массив

Задания, оценивающие сформированную компетенцию ПК-1.3

9. В какой из следующих строк выполняется обращение к адресу седьмого элемента массива в памяти?
   a) mas[7]
   b) mas+6
   c) mas(7)
   d) mas[6]
10. Чему будет равна переменная a, после выполнения этого кода:
```
int a;
for(a = 0; a < 10; a++);
```
   a) 1
   b) 10
   c) 0
   d) 9
11. Что означает понятие «исключительная ситуация»?
   a) способ генерирования специального кода ошибок
   b) нештатное событие, влияющее на дальнейшее выполнение программы
   c) особая ситуация, которая возникает при вызове метода класса
   d) автоматическое исправление ошибок
12. Какой из ниже перечисленных операторов, не является циклом в C++?
   a) for()
   b) do{}while()
   c) repeat{}until()
   d) while()

Задания, оценивающие сформированную компетенцию ПК-1.4

13. До каких пор будут выполняться операторы в теле цикла while (x < 100){}?
   a) Пока x равен 100
   b) Пока x меньше или равен 100
   c) Пока x больше или равен 100
   d) Пока x строго меньше 100
14. Укажите тип возвращаемого значения следующей функции:
```
double func(char x, float v, int t);
```
   a) int
   b) float
   c) char
   d) double
15. Что делает операция ++?
   a) Увеличивает значение переменной на единицу
   b) Увеличивает значение переменной на два
   c) Сдвигает значение в массиве вперед на один индекс

d) В языке C++ такой операции не существует

16. Этап преобразования исходного текста программы, написанного на языке программирования, в язык, в ассемблерный код называется…
    a) Генерацией
    b) Компиляцией
    c) Ассемблированием
    d) Препроцессингом

Задания, оценивающие сформированную компетенцию ПК-1.5

17. Скрытие полей объекта с целью обеспечения доступа к ним только посредством методов класса называют…
    a) Абстрагированием
    b) Делегированием
    c) Инкапсуляцией
    d) Виртуализацией
18. Какой из следующих классов обрабатывает процесс записи в файл?
    a) input_file
    b) ifstream
    c) ofstream
    d) FILE
19. С помощью чего реализуется принцип полиморфизма в C++?
    a) наличия множественного наследования.
    b) наличия виртуальных методов.
    c) Наличия прав доступа к компонентам класса.
    d) наличия абстрактных классов.
20. Какая операция позволяет получить значение переменной, записанное по адресу, который содержится в указателе?
    a) *
    b) ?
    c) &
    d) ->

Задания, оценивающие сформированную компетенцию ПК-1.6

21. Цикл с постусловием?
    a) do while
    b) for
    c) while
    d) repeat until
22. Тело любого цикла выполняется до тех пор, пока его условие ...
    a) ложно
    b) у цикла нет условия
    c) истинно
    d) цикл выполняется бесконечно, пока не будет вызван *break* внутри цикла
23. Как называется функция, которая вызывает саму себя?
    a) Рекурсивной
    b) Дружественной
    c) Конструктором
    d) Деструктором

24. Какая функция, не будучи компонентом класса, имеет доступ к его защищенным внутренним компонентам?
    a) Рекурсивная.
    b) Шаблонная.
    c) Полиморфная.
    d) Дружеская.

4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИИ

4.1. Система оценивания компетенций

Система оценивания компетенций, формируемых в ходе освоения дисциплины, основана на оценивании всех видов работ.

В таблицах 4.1.1-4.1.3 представлены критерии оценивания каждого вида работ.

Таблица 4.1.1 – Критерии оценивания лабораторной работы

Уровень усвоения	Описание
Отлично	Правильное выполнение задания и точные ответы на все контрольные и дополнительные вопросы. Сформированные умения по применению полученных знаний в решении профессиональных задач в рамках учебного материала.
Хорошо	Выполнение задания с небольшими неточностями и правильные ответы на большинство контрольных и дополнительных вопросов. В целом успешные, но содержащие отдельные пробелы умения по применению полученных знаний в решении профессиональных задач в рамках усвоенного учебного материала.
Удовлетворительно	Выполнение задания с существенными неточностями и ответы с затруднениями на большинство контрольных и дополнительных вопросов. В целом успешные, но не систематически осуществляемые умения по применению полученных знаний в решении профессиональных задач в рамках усвоенного учебного материала.
Неудовлетворительно	Неправильное выполнение задания, множество неточностей в ответах на контрольные и дополнительные вопросы. Частично освоенные или полностью неосвоенные умения по применению полученных знаний в решении профессиональных задач в рамках учебного материала.

Таблица 4.1.2 – Критерии оценки дифференцированного зачета (5 семестр)

Уровень усвоения	Описание
Отлично	Глубокие исчерпывающие знания программного материала, последовательные, содержательные, полные, правильные и конкретные ответы на теоретический и практический вопросы билета и дополнительные вопросы преподавателя. Написан код программы, который выполняет все поставленные задачи и работает без ошибок.
Хорошо	Твердые и достаточно полные знания всего программного материала, последовательные, правильные, конкретные ответы на поставленные вопросы при свободном устранении замечаний по отдельным вопросам. Написан код программы, который выполняет почти все поставленные задачи или во время работы в некоторых ситуациях возникают ошибки, но все задачи выполняются.
Удовлетворительно	Знание и понимание основного программного материала; правильные, без грубых ошибок ответы на поставленные вопросы при устранении неточностей и несущественных ошибок в освещении отдельных положений при наводящих вопросах преподавателя. Написан код программы, который выполняет только часть задач или программа работает со сбоями из-за значительных недоработок.

Неудовлетворительно	Грубые ошибки в ответе, непонимание сущности излагаемых вопросов; неуверенные и неточные ответы на дополнительные вопросы. Программа не написана.

Таблица 4.1.3 – Критерии оценки экзамена (6 семестр)

Уровень усвоения	Описание
Отлично	Глубокие исчерпывающие знания всего программного материала, последовательные, содержательные, полные, правильные и конкретные ответы на оба вопроса экзаменационного билета и дополнительные вопросы преподавателя.
Хорошо	Твердые и достаточно полные знания всего программного материала, последовательные, правильные, конкретные ответы на поставленные вопросы при свободном устранении замечаний по отдельным вопросам.
Удовлетворительно	Знание и понимание основного программного материала; правильные, без грубых ошибок ответы на поставленные вопросы при устранении неточностей и несущественных ошибок в освещении отдельных положений при наводящих вопросах преподавателя.
Неудовлетворительно	Грубые ошибки в ответе, непонимание сущности излагаемых вопросов; неуверенные и неточные ответы на дополнительные вопросы.

4.2. Итоговая оценка по дисциплине

Оценка уровня усвоения компетенций производится на основе средней оценки по всем видам работ: лабораторных работ, дифференцированного зачета, экзамена.

Оценки ставятся по 5-балльной шкале. Округление оценки производится в пользу студента. Итоговая оценка выставляется в ведомость согласно следующему правилу:

Итоговая оценка по дисциплине			
Неудовлетворительно	**Удовлетворительно**	**Хорошо**	**Отлично**
0 – 2.9	3.0-3.49	3.5-4.4	4.5-5.0

МИНОБРНАУКИ РОССИИ
Восточно-Сибирский государственный университет технологий и управления
Технологический колледж
Кафедра «Системы информатики»

Методические рекомендации по организации изучения дисциплин профессионального модуля

ПМ.01«Разработка программных модулей программного обеспечения для компьютерных систем»

для специальности: 09.02.03 – Программирование в компьютерных системах

Форма обучения: очная

Год набора: 2018

Улан-Удэ
2018

Организация образовательного процесса регламентируется учебным планом и расписанием учебных занятий. Язык обучения (преподавания) - русский. Для всех видов аудиторных занятий академический час устанавливается продолжительностью 45 минут.

При формировании своей индивидуальной образовательной траектории обучающийся имеет право на перезачет соответствующих дисциплин и профессиональных модулей, освоенных в процессе предшествующего обучения, который освобождает обучающегося от необходимости их повторного освоения.

1. Образовательные технологии

Учебный процесс при преподавании курса основывается на использовании традиционных, инновационных и информационных образовательных технологий. Традиционные образовательные технологии представлены лекциями и семинарскими (практическими) занятиями. Инновационные образовательные технологии используются в виде широкого применения активных и интерактивных форм проведения занятий. Информационные образовательные технологии реализуются путем активизации самостоятельной работы студентов в информационной образовательной среде.

2. Лекции

Лекционный курс предполагает систематизированное изложение основных вопросов учебного плана.

На первой лекции лектор обязан предупредить студентов, применительно к какому базовому учебнику (учебникам, учебным пособиям) будет прочитан курс.

Лекционный курс должен давать наибольший объем информации и обеспечивать более глубокое понимание учебных вопросов при значительно меньшей затрате времени, чем это требуется большинству студентов на самостоятельное изучение материала.

3. Лабораторные работы

Лабораторные работы являются практической частью дисциплины и имеют целью на практике подтвердить основные положения теории с целью закрепления изученного материала.

Работы выполняются индивидуально каждым студентом после прослушивания курса лекций по данной работе. Приступая к выполнению лабораторных занятий, студент должен изучить основные положения теории к данной работе, описание работы из методического указания.

Лабораторные занятия выполняются согласно порядку выполнения работ в методическом указании.

Выполненные работы защищаются, и их положительная оценка является одним из необходимых условий для получения допуска к промежуточной аттестации.

Отчеты по лабораторным занятиям оформляются на листах писчей бумаги формата А4 каждым обучающимся. В отчете должны быть представлены:

1) Титульный лист.

2) Название и цель выполняемой работы.

3) Краткое описание хода работы.

4) Ответы на контрольные вопросы.

5) Выводы, отражающие цель и результаты выполнения работы.

Рисунки помещаются по тексту в соответствующих местах или на отдельных листах и выполняются в удобном для чтения масштабе.

Работа должна быть подписана и датирована студентом.

Выводы к лабораторным работам должны отражать цель выполнения работы, краткий ход выполнения работы с главным результатом, анализ результатов в соответствии с целью работы.

4. Самостоятельная работа студента

Для успешного усвоения курса необходимо не только посещать аудиторные занятия, но и вести активную самостоятельную работу. Самостоятельная работа включает проработка и закрепление лекционного материала, подготовка к выполнению и к защите лабораторных работ.

4.1. Закрепление лекционного материала

При самостоятельной проработке курса обучающиеся должны:

– просматривать основные определения и факты;

– повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;

– изучить рекомендованную основную и дополнительную литературу, составлять тезисы, аннотации и конспекты наиболее важных моментов;

– самостоятельно выполнять задания, аналогичные предлагаемым на занятиях;

– использовать для самопроверки материалы фонда оценочных средств.

4.2. Подготовка к лабораторным работам

При подготовке к лабораторным занятиям необходимо изучить основные теоретические положения к лабораторной работе.

В случае если работа полностью не выполнена на занятии, ее можно завершить самостоятельно дома и результаты продемонстрировать преподавателю на занятиях. Оформить отчет по лабораторной работе, руководствуясь требованиями, представленными в методических указаниях к лабораторным работам.

При подготовке к защите лабораторной работы необходимо ответить на контрольные вопросы.

4.3. Работа с медиаматериалами

Самостоятельная работа в современном учебном процессе подразумевает ознакомление студента с различными видео и аудиоматериалами на русском и иностранных языках. Это позволит усилить запоминание теоретических положений через визуальное и слуховое восприятие, ознакомиться с авторским изложением сложных моментов, сформировать свою точку зрения с учетом представленных дискуссий, разобрать примеры и практические кейсы, выполнить задания и отвечать на поставленные вопросы.

5. Курсовой проект

В курсе используются исследовательские методы обучения, предполагающие самостоятельный творческий поиск и применение знаний обучающимся. Курсовой проект – это письменная работа, которая строится по логике проведения классического научного исследования.

Целью проекта является повышение уровня профессиональной подготовки обучающегося. Проект формирует следующие компетенции:

– усвоение теоретического материала и путей его применения на практике;

– навыки творческого мышления;

– воспитание чувства ответственности за качество принятых решений;

– навык самостоятельной профессиональной деятельности;

– комплексная работа со специальной литературой и информационными ресурсами;

– научно-исследовательская деятельность.

Проект входит в индивидуальное портфолио обучающегося.

В случае наличия существенных замечаний руководителя работа возвращается обучающемуся на доработку.

Допускается открытая защита в присутствии всей учебной группы. Вопросы, задаваемые автору проекта, не должны выходить за рамки тематики проекта. При своевременной защите работа оценивается наивысшим баллом, при опоздании на 1 неделю балл снижается на 2, при опоздании на 2 недели балл снижается еще раз на 2. При опоздании более чем на 2 недели работа не оценивается.

6. Оценивание по дисциплине

Оценка уровня усвоения компетенций производится на основе средней оценки по всем видам работ:

- лабораторные работы;
- курсовой проект;
- промежуточная аттестация.

Оценки ставятся по 5-балльной шкале. Округление оценки производится в пользу студента. Итоговая оценка выставляется в ведомость согласно следующему правилу:

Итоговая оценка по дисциплине			
Неудовлетворительно 2	**Удовлетворительно 3**	**Хорошо 4**	**Отлично 5**
0 – 2.9	3.0-3.49	3.5-4.4	4.5-5.0

7. Методические рекомендации по обучению лиц с ограниченными возможностями здоровья и инвалидов

Профессорско-педагогический состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями здоровья, индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания тьюторами, психологами, социальными работниками, прошедшими подготовку ассистентами.

В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. N АК-44/05вн) в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производятся с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.

Медиаматериалы также следует использовать и адаптировать с учетом индивидуальных особенностей обучения лиц с ОВЗ.

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ.

Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей. Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

– в печатной или электронной форме (для лиц с нарушениями опорно-двигательного аппарата);

– в печатной форме или электронной форме с увеличенным шрифтом и контрастностью (для лиц с нарушениями слуха, речи, зрения);

– методом чтения ассистентом задания вслух (для лиц с нарушениями зрения).

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

– письменно на бумаге или набором ответов на компьютере (для лиц с нарушениями слуха, речи);

– выбором ответа из возможных вариантов с использованием услуг ассистента (для лиц с нарушениями опорно-двигательного аппарата);

– устно (для лиц с нарушениями зрения, опорно-двигательного аппарата).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

МИНОБРНАУКИ РОССИИ
Восточно-Сибирский государственный университет технологий и управления
Технологический колледж
Кафедра «Системы информатики»

Методические рекомендации по организации изучения дисциплины

«Системное программирование»

для специальности: 09.02.03 – Программирование в компьютерных системах

Форма обучения: очная

Год набора: 2018

Улан-Удэ
2018

Организация образовательного процесса регламентируется учебным планом и расписанием учебных занятий. Язык обучения (преподавания) - русский. Для всех видов аудиторных занятий академический час устанавливается продолжительностью 45 минут.

При формировании своей индивидуальной образовательной траектории обучающийся имеет право на перезачет соответствующих дисциплин и профессиональных модулей, освоенных в процессе предшествующего обучения, который освобождает обучающегося от необходимости их повторного освоения.

1. Образовательные технологии

Учебный процесс при преподавании курса основывается на использовании традиционных, инновационных и информационных образовательных технологий. Традиционные образовательные технологии представлены лекциями и семинарскими (практическими) занятиями. Инновационные образовательные технологии используются в виде широкого применения активных и интерактивных форм проведения занятий. Информационные образовательные технологии реализуются путем активизации самостоятельной работы студентов в информационной образовательной среде.

2. Лекции

Лекционный курс предполагает систематизированное изложение основных вопросов учебного плана.

На первой лекции лектор обязан предупредить студентов, применительно к какому базовому учебнику (учебникам, учебным пособиям) будет прочитан курс.

Лекционный курс должен давать наибольший объем информации и обеспечивать более глубокое понимание учебных вопросов при значительно меньшей затрате времени, чем это требуется большинству студентов на самостоятельное изучение материала.

3. Лабораторные работы

Лабораторные работы являются практической частью дисциплины и имеют целью на практике подтвердить основные положения теории с целью закрепления изученного материала.

Работы выполняются индивидуально каждым студентом после прослушивания курса лекций по данной работе. Приступая к выполнению лабораторных занятий, студент должен изучить основные положения теории к данной работе, описание работы из методического указания.

Лабораторные занятия выполняются согласно порядку выполнения работ в методическом указании.

Выполненные работы защищаются, и их положительная оценка является одним из необходимых условий для получения допуска к промежуточной аттестации.

Отчеты по лабораторным занятиям оформляются на листах писчей бумаги формата А4 каждым обучающимся. В отчете должны быть представлены:

1) Титульный лист.

2) Название и цель выполняемой работы.

3) Краткое описание хода работы.

4) Ответы на контрольные вопросы.

5) Выводы, отражающие цель и результаты выполнения работы.

Рисунки помещаются по тексту в соответствующих местах или на отдельных листах и выполняются в удобном для чтения масштабе.

Работа должна быть подписана и датирована студентом.

Выводы к лабораторным работам должны отражать цель выполнения работы, краткий ход выполнения работы с главным результатом, анализ результатов в соответствии с целью работы.

4. Самостоятельная работа студента

Для успешного усвоения курса необходимо не только посещать аудиторные занятия, но и вести активную самостоятельную работу. Самостоятельная работа включает проработка и закрепление лекционного материала, подготовка к выполнению и к защите лабораторных работ.

4.1. Закрепление лекционного материала

При самостоятельной проработке курса обучающиеся должны:

– просматривать основные определения и факты;

– повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;

– изучить рекомендованную основную и дополнительную литературу, составлять тезисы, аннотации и конспекты наиболее важных моментов;

– самостоятельно выполнять задания, аналогичные предлагаемым на занятиях;

– использовать для самопроверки материалы фонда оценочных средств.

4.2. Подготовка к лабораторным работам

При подготовке к лабораторным занятиям необходимо изучить основные теоретические положения к лабораторной работе.

В случае если работа полностью не выполнена на занятии, ее можно завершить самостоятельно дома и результаты продемонстрировать преподавателю на занятиях. Оформить отчет по лабораторной работе, руководствуясь требованиями, представленными в методических указаниях к лабораторным работам.

При подготовке к защите лабораторной работы необходимо ответить на контрольные вопросы.

4.3. Работа с медиаматериалами

Самостоятельная работа в современном учебном процессе подразумевает ознакомление студента с различными видео и аудиоматериалами на русском и иностранных языках. Это позволит усилить запоминание теоретических положений через визуальное и слуховое восприятие, ознакомиться с авторским изложением сложных моментов, сформировать свою точку зрения с учетом представленных дискуссий, разобрать примеры и практические кейсы, выполнить задания и отвечать на поставленные вопросы.

5. Курсовой проект

В курсе используются исследовательские методы обучения, предполагающие самостоятельный творческий поиск и применение знаний обучающимся. Курсовой проект – это письменная работа, которая строится по логике проведения классического научного исследования.

Целью проекта является повышение уровня профессиональной подготовки обучающегося. Проект формирует следующие компетенции:

– усвоение теоретического материала и путей его применения на практике;

– навыки творческого мышления;

– воспитание чувства ответственности за качество принятых решений;

– навык самостоятельной профессиональной деятельности;

– комплексная работа со специальной литературой и информационными ресурсами;

– научно-исследовательская деятельность.

Проект входит в индивидуальное портфолио обучающегося.

В случае наличия существенных замечаний руководителя работа возвращается обучающемуся на доработку.

Допускается открытая защита в присутствии всей учебной группы. Вопросы, задаваемые автору проекта, не должны выходить за рамки тематики проекта. При своевременной защите работа оценивается наивысшим баллом, при опоздании на 1 неделю балл снижается на 2, при опоздании на 2 недели балл снижается еще раз на 2. При опоздании более чем на 2 недели работа не оценивается.

6. Оценивание по дисциплине

Оценка уровня усвоения компетенций производится на основе средней оценки по всем видам работ:

- лабораторные работы;
- курсовой проект;
- промежуточная аттестация.

Оценки ставятся по 5-балльной шкале. Округление оценки производится в пользу студента. Итоговая оценка выставляется в ведомость согласно следующему правилу:

Итоговая оценка по дисциплине			
Неудовлетворительно 2	**Удовлетворительно 3**	**Хорошо 4**	**Отлично 5**
0 – 2.9	3.0-3.49	3.5-4.4	4.5-5.0

7. Методические рекомендации по обучению лиц с ограниченными возможностями здоровья и инвалидов

Профессорско-педагогический состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями здоровья, индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания тьюторами, психологами, социальными работниками, прошедшими подготовку ассистентами.

В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. N АК-44/05вн) в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производятся с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.

Медиаматериалы также следует использовать и адаптировать с учетом индивидуальных особенностей обучения лиц с ОВЗ.

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ.

Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей. Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

– в печатной или электронной форме (для лиц с нарушениями опорно-двигательного аппарата);

– в печатной форме или электронной форме с увеличенным шрифтом и контрастностью (для лиц с нарушениями слуха, речи, зрения);

– методом чтения ассистентом задания вслух (для лиц с нарушениями зрения).

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

– письменно на бумаге или набором ответов на компьютере (для лиц с нарушениями слуха, речи);

– выбором ответа из возможных вариантов с использованием услуг ассистента (для лиц с нарушениями опорно-двигательного аппарата);

– устно (для лиц с нарушениями зрения, опорно-двигательного аппарата).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

МИНОБРНАУКИ РОССИИ
Восточно-Сибирский государственный университет технологий и управления
Технологический колледж
Кафедра «Системы информатики»

Методические рекомендации по организации изучения дисциплины

«Прикладное программирование»

для специальности: 09.02.03 – Программирование в компьютерных системах

Форма обучения: очная

Год набора: 2018

Улан-Удэ
2018

Организация образовательного процесса регламентируется учебным планом и расписанием учебных занятий. Язык обучения (преподавания) - русский. Для всех видов аудиторных занятий академический час устанавливается продолжительностью 45 минут.

При формировании своей индивидуальной образовательной траектории обучающийся имеет право на перезачет соответствующих дисциплин и профессиональных модулей, освоенных в процессе предшествующего обучения, который освобождает обучающегося от необходимости их повторного освоения.

1. Образовательные технологии

Учебный процесс при преподавании курса основывается на использовании традиционных, инновационных и информационных образовательных технологий. Традиционные образовательные технологии представлены лекциями и семинарскими (практическими) занятиями. Инновационные образовательные технологии используются в виде широкого применения активных и интерактивных форм проведения занятий. Информационные образовательные технологии реализуются путем активизации самостоятельной работы студентов в информационной образовательной среде.

2. Лекции

Лекционный курс предполагает систематизированное изложение основных вопросов учебного плана.

На первой лекции лектор обязан предупредить студентов, применительно к какому базовому учебнику (учебникам, учебным пособиям) будет прочитан курс.

Лекционный курс должен давать наибольший объем информации и обеспечивать более глубокое понимание учебных вопросов при значительно меньшей затрате времени, чем это требуется большинству студентов на самостоятельное изучение материала.

3. Лабораторные работы

Лабораторные работы являются практической частью дисциплины и имеют целью на практике подтвердить основные положения теории с целью закрепления изученного материала.

Работы выполняются индивидуально каждым студентом после прослушивания курса лекций по данной работе. Приступая к выполнению лабораторных занятий, студент должен изучить основные положения теории к данной работе, описание работы из методического указания.

Лабораторные занятия выполняются согласно порядку выполнения работ в методическом указании.

Выполненные работы защищаются, и их положительная оценка является одним из необходимых условий для получения допуска к промежуточной аттестации.

Отчеты по лабораторным занятиям оформляются на листах писчей бумаги формата А4 каждым обучающимся. В отчете должны быть представлены:

1) Титульный лист.

2) Название и цель выполняемой работы.

3) Краткое описание хода работы.

4) Ответы на контрольные вопросы.

5) Выводы, отражающие цель и результаты выполнения работы.

Рисунки помещаются по тексту в соответствующих местах или на отдельных листах и выполняются в удобном для чтения масштабе.

Работа должна быть подписана и датирована студентом.

Выводы к лабораторным работам должны отражать цель выполнения работы, краткий ход выполнения работы с главным результатом, анализ результатов в соответствии с целью работы.

4. Самостоятельная работа студента

Для успешного усвоения курса необходимо не только посещать аудиторные занятия, но и вести активную самостоятельную работу. Самостоятельная работа включает проработка и закрепление лекционного материала, подготовка к выполнению и к защите лабораторных работ.

4.1. Закрепление лекционного материала

При самостоятельной проработке курса обучающиеся должны:

– просматривать основные определения и факты;

– повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;

– изучить рекомендованную основную и дополнительную литературу, составлять тезисы, аннотации и конспекты наиболее важных моментов;

– самостоятельно выполнять задания, аналогичные предлагаемым на занятиях;

– использовать для самопроверки материалы фонда оценочных средств.

4.2. Подготовка к лабораторным работам

При подготовке к лабораторным занятиям необходимо изучить основные теоретические положения к лабораторной работе.

В случае если работа полностью не выполнена на занятии, ее можно завершить самостоятельно дома и результаты продемонстрировать преподавателю на занятиях. Оформить отчет по лабораторной работе, руководствуясь требованиями, представленными в методических указаниях к лабораторным работам.

При подготовке к защите лабораторной работы необходимо ответить на контрольные вопросы.

4.3. Работа с медиаматериалами

Самостоятельная работа в современном учебном процессе подразумевает ознакомление студента с различными видео и аудиоматериалами на русском и иностранных языках. Это позволит усилить запоминание теоретических положений через визуальное и слуховое восприятие, ознакомиться с авторским изложением сложных моментов, сформировать свою точку зрения с учетом представленных дискуссий, разобрать примеры и практические кейсы, выполнить задания и отвечать на поставленные вопросы.

5. Оценивание по дисциплине

Оценка уровня усвоения компетенций производится на основе средней оценки по всем видам работ:

– лабораторные работы;

– промежуточная аттестация.

Оценки ставятся по 5-балльной шкале. Округление оценки производится в пользу студента. Итоговая оценка выставляется в ведомость согласно следующему правилу:

Итоговая оценка по дисциплине			
Неудовлетворительно 2	**Удовлетворительно 3**	**Хорошо 4**	**Отлично 5**
0 – 2.9	3.0-3.49	3.5-4.4	4.5-5.0

6. Методические рекомендации по обучению лиц с ограниченными возможностями здоровья и инвалидов

Профессорско-педагогический состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями

здоровья, индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания тьюторами, психологами, социальными работниками, прошедшими подготовку ассистентами.

В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. N АК-44/05вн) в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производятся с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.

Медиаматериалы также следует использовать и адаптировать с учетом индивидуальных особенностей обучения лиц с ОВЗ.

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ.

Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей. Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

– в печатной или электронной форме (для лиц с нарушениями опорно-двигательного аппарата);

– в печатной форме или электронной форме с увеличенным шрифтом и контрастностью (для лиц с нарушениями слуха, речи, зрения);

– методом чтения ассистентом задания вслух (для лиц с нарушениями зрения).

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

– письменно на бумаге или набором ответов на компьютере (для лиц с нарушениями слуха, речи);

– выбором ответа из возможных вариантов с использованием услуг ассистента (для лиц с нарушениями опорно-двигательного аппарата);

– устно (для лиц с нарушениями зрения, опорно-двигательного аппарата).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.