

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
"Амурский государственный университет"

УТВЕРЖДАЮ

Проректор по учебной и научной
работе

Лейфа А.В. Лейфа

« 2 » марта 2024 г.

РАБОЧАЯ ПРОГРАММА

МДК

МДК.01.01 Разработка программных модулей

Специальность 09.02.07 Информационные системы и программирование

Квалификация выпускника – Программист

Год набора – 2024

Курс 1,2 Семестр 2,3

Экзамен 3 сем

Дифференцированный зачет 2 сем

Общая трудоемкость МДК 234.0 (академ. час)

Составитель Н.С. Кузнецова, Преподаватель,

Факультет среднего профессионального образования

ЦМК инженерно-технических и информационных дисциплин

Рабочая программа составлена на основании Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного приказом Министерства образования и науки Российской Федерации от 09.12.2016 № 1547

Рабочая программа обсуждена на заседании кафедры инженерно-технических и информационных дисциплин

09.02.2024 г. , протокол № 6

Заведующий кафедрой _____

СОГЛАСОВАНО

Зам. декана по учебной работе

Кирилюк Н.В. Кирилюк

« 2 » марта 2024 г.

СОГЛАСОВАНО

Выпускающая кафедра

Казакова Т.А. Казакова

« 2 » марта 2024 г.

СОГЛАСОВАНО

Научная библиотека

Петрович О.В. Петрович

« 2 » марта 2024 г.

СОГЛАСОВАНО

Центр цифровой трансформации и
технического обеспечения

Годосейчук А.А. Годосейчук

« 2 » марта 2024 г.

1. ОБЛАСТЬ ПРИМЕНЕНИЯ

Рабочая программа профессионального модуля ПМ.01 Разработка модулей программного обеспечения для компьютерных систем (далее программа) является составной частью основной профессиональной образовательной программы в соответствии с федеральным государственным образовательным стандартом среднего профессионального образования (далее – ФГОС СПО) по специальности 09.02.07 Информационные системы и программирование в части освоения основного вида деятельности: разработка модулей программного обеспечения для компьютерных систем и соответствующих ему общих компетенций и профессиональных компетенций.

2. МЕСТО МДК В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

МДК входит в профессиональный цикл ПМ.01 Разработка модулей программного обеспечения для компьютерных систем по учебному плану, читается во 2,3 семестре в объеме 234 академических часов. Для успешного освоения МДК Разработка программных модулей специальности 09.02.07 Информационные системы и программирование, обучающиеся должны владеть компетенциями, полученными при изучении дисциплин: ОП.01. Операционные системы и среды; ОП.03. Информационные технологии; ОП.04 Основы алгоритмизации и программирования. На компетенциях, формируемых, дисциплиной Разработка программных модулей базируются дисциплины МДК.01.02 Поддержка и тестирование программных модулей, МДК.01.03 Разработка мобильных приложений, МДК.01.04 Системное программирование, учебная и производственная практика, преддипломная практика, а также подготовка и защита выпускной квалификационной работы.

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ МДК И ИНДИКАТОРЫ ИХ ДОСТИЖЕНИЯ

3.1. Профессиональные компетенции и индикаторы их достижения

| Категория (группа) профессиональных компетенций | Код и наименование профессиональных компетенции | Минимальные требования |
|---|---|---|
| ПК 1.1. | ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. | Практический опыт: Разрабатывать алгоритм решения поставленной задачи и реализовывать его средствами автоматизированного проектирования. Умения: Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. Оформлять документацию на программные средства. Оценка сложности алгоритма. Знания: Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования. Актуальная нормативно- правовая база в области документирования алгоритмов. |

| | | |
|---------|---|---|
| ПК 1.2. | ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием. | <p>Практический опыт: Разрабатывать код программного продукта на основе готовой спецификации на уровне модуля. Разрабатывать мобильные приложения.</p> <p>Умения: Создавать программу по разработанному алгоритму как отдельный модуль. Оформлять документацию на программные средства. Осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.</p> <p>Знания: Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования. Знание API современных мобильных операционных систем.</p> |
| ПК 1.3. | ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств | <p>Практический опыт: Использовать инструментальные средства на этапе отладки программного продукта. Проводить тестирование программного модуля по определенному сценарию.</p> <p>Умения: Выполнять отладку и тестирование программы на уровне модуля. Оформлять документацию на программные средства. Применять инструментальные средства отладки программного обеспечения.</p> <p>Знания: Основные принципы отладки и тестирования программных продуктов. Инструментарий отладки программных продуктов.</p> |

4. СТРУКТУРА МДК

Общая трудоемкость МДК составляет 6.50 зачетных единицы, 234.0 академических часов.

1 – № п/п

2 – Тема (раздел) МДК, курсовая работа (проект), промежуточная аттестация

3 – Семестр

4 – Виды контактной работы и трудоемкость (в академических часах)

4.1 – Л (Лекции)

4.2 – Лекции в виде практической подготовки

4.3 – ПЗ (Практические занятия)

4.4 – Практические занятия в виде практической подготовки

4.5 – ЛР (Лабораторные работы)

4.6 – Лабораторные работы в виде практической подготовки

4.7 – ИКР (Иная контактная работа)

4.8 – КТО (Контроль теоретического обучения)

4.9 – КЭ (Контроль на экзамене)

5 – Контроль (в академических часах)

6 – Самостоятельная работа (в академических часах)

7 – Формы текущего контроля успеваемости

4.10 – У (Уроки)

4.11 – С (Семинарские занятия)

| 1 | 2 | 3 | 4 | | | | | | | | | | | 5 | 6 | 7 |
|---|--|------|-----|-----|-----|-----|-----|-----|------|------|-----|-----|-----|---|---|--|
| | | | 4.1 | 4.2 | 4.3 | 4.4 | 4.5 | 4.6 | 4.10 | 4.11 | 4.7 | 4.8 | 4.9 | | | |
| 1 | Тема 1. Жизненный цикл ПО | 2 | 6 | | | | | | | | | | | | 4 | устный опрос |
| 2 | Тема 2. Структурное программирование | 2 | 10 | | 16 | | 8 | | | | | | | | 4 | устный опрос, оценка выполнения практических, лабораторных работ |
| 3 | Тема 3. Объектно-ориентированное программирование | 2 | 10 | | 20 | | 8 | | | | | | | 2 | 6 | устный опрос, оценка выполнения практических, лабораторных работ |
| 4 | Тема 4. Паттерны проектирования | 2 | 8 | | 16 | | 4 | | | | | | | | 4 | устный опрос, оценка выполнения практических, лабораторных работ |
| 5 | Тема 5. Событийно- | 2, 3 | 8 | | 20 | | 6 | | | | | | | | 6 | устный опрос, |

| | | | | | | | | | | | | | | | | | | |
|---|--|---|------|--|------|--|------|--|-----|-----|-----|-----|-----|-----|------|--|--|--|
| | управляемое программирование | | | | | | | | | | | | | | | | | оценка выполнения практических, лабораторных работ |
| 6 | Тема 6. Оптимизация и рефакторинг кода | 3 | 8 | | | | 6 | | | | | | | | 4 | | | устный опрос, оценка выполнения лабораторных работ |
| 7 | Тема 7. Разработка пользовательского интерфейса. | 3 | 10 | | | | 10 | | | | | | 2 | 6 | | | | устный опрос, оценка выполнения лабораторных работ |
| 8 | Тема 8. Службы доступа к данным | 3 | 8 | | | | 6 | | | | 2 | | | 4 | | | | устный опрос, оценка выполнения лабораторных работ |
| 9 | Экзамен | 3 | | | | | | | | | | 2 | | | | | | экзамен |
| | Итого | | 68.0 | | 72.0 | | 48.0 | | 0.0 | 0.0 | 0.0 | 2.0 | 2.0 | 4.0 | 38.0 | | | |

5. СОДЕРЖАНИЕ МДК

5.1. Лекции

| № п/п | Наименование темы (раздела) | Содержание темы (раздела) |
|-------|---|---|
| 1 | Тема 1. Жизненный цикл ПО | 1.Понятие ЖЦ ПО. Этапы ЖЦ ПО 2.Технология структурного программирования. 3.Инструментальные средства оформления и документирования алгоритмов программ |
| 2 | Тема 2. Структурное программирование | 4.Оценка сложности алгоритма: классификация, классы алгоритмов, неразрешимые задачи 5.Системы контроля версий: виды, принципы организации работы 6.Нормативно- правовая база в области документирования алгоритмов. 7.Оценка сложности алгоритма: классификация, классы алгоритмов, неразрешимые задачи 8.Типовые алгоритмы обработки массивов, рекурсии и т.д. |
| 3 | Тема 3. Объектно-ориентированное программирование | 9.Основные принципы объектно-ориентированного программирования. Классы: основные понятия. 10.Перегрузка методов. Операции класса. Иерархия классов. |

| | | |
|---|--|--|
| | | 11.Синтаксис интерфейсов. Интерфейсы и наследование. 12.Структуры. Делегаты. Регулярные выражения 13.Параметризованные классы. |
| 4 | Тема 4. Паттерны проектирования | 14.Назначение и виды паттернов. 15.Основные шаблоны. 16.Порождающие шаблоны. 17.Структурные шаблоны. |
| 5 | Тема5. Событийно-управляемое программирование | 18.Событийно-управляемое программирование. 19.Элементы управления. 20.Диалоговые окна. Обработчики событий. 21.Введение в графику. |
| 6 | Тема 6. Оптимизация и рефакторинг кода | 22.Способы оптимизации программного кода. 23.Методы оптимизации программного кода. 24.Методы организации рефакторинга и оптимизации кода. 25.Принципы работы с системой контроля версии |
| 7 | Тема 7. Разработка пользовательского интерфейса. | 26.Основные принципы разработки пользовательского интерфейса 27.Стандартизация пользовательского интерфейса. Виды пользовательского интерфейса 28.Проектирование пользовательских интерфейсов 29.Разработка сценария диалога. Методы разработки гибкого интерфейса 30.Визуальные атрибуты отображаемой информации |
| 8 | Тема 8. Службы доступа к данным | 31.Работа с базами данных. Основные способы доступа к данным 32.Организация доступа к данным: подключенный режим, автономный режим, технология Entity Framework 33.Создание таблиц, отчетов, работа с записями. 34.Создание хранимых процедур |

5.2. Практические занятия

| Наименование темы | Содержание темы |
|-----------------------------|---|
| Практическая работа № 1 -2 | № 1 -2 .Оценка сложности алгоритмов сортировки. |
| Практическая работа № 3-4 | № 3-4.Оценка сложности алгоритмов поиска |
| Практическая работа № 5-6 | № 5-6.Оценка сложности рекурсивных алгоритмов |
| Практическая работа № 7-8 | № 7-8.Оценка сложности эвристических алгоритмов |
| Практическая работа № 9 | № 9.Работа с классами |
| Практическая работа № 10 | № 10.Перегрузка методов. |
| Практическая работа № 11 | № 11.Определение операций в классе |
| Практическая работа № 12 | № 12.Создание наследованных классов |
| Практическая работа № 13 | № 13.Работа с объектами через интерфейсы |
| Практическая работа № 14 | № 14.Использование стандартных интерфейсов |
| Практическая работа № 15 | № 15. Работа с типом данных структура |
| Практическая работа № 16-17 | № 16-17.Коллекции. Параметризованные классы |

| | |
|-----------------------------|--|
| Практическая работа № 18 | № 18.Использование регулярных выражений |
| Практическая работа № 19-20 | № 19-20.Использование основных шаблонов |
| Практическая работа № 21-22 | № 21-22.Использование порождающих шаблонов |
| Практическая работа № 23-24 | № 23-24.Использование структурных шаблонов |
| Практическая работа № 25-26 | № 25-26.Использование поведенческих шаблонов |
| Практическая работа № 27-28 | № 27-28.Разработка приложения с использованием текстовых компонентов |
| Практическая работа № 29-30 | № 29-30.Разработка приложения с несколькими формами |
| Практическая работа № 31-32 | № 31-32.Разработка приложения с не визуальными компонентами |
| Практическая работа № 33-34 | № 33-34.Разработка игрового приложения |
| Практическая работа № 35-36 | № 35-36.Разработка приложения с анимацией |

5.3. Лабораторные занятия

| Наименование темы | Содержание темы |
|-----------------------------|---|
| Лабораторное занятие №1 | №1.Оценка сложности рекурсивных алгоритмов. |
| Лабораторное занятие №2 | №2.Оценка сложности эвристических алгоритмов. |
| Лабораторное занятие №3 | №3.Использование стандартных интерфейсов. |
| Лабораторное занятие №4 | №4.Работа с типом данных структура. |
| Лабораторное занятие №5-6 | №5-6.Коллекции. Параметризованные классы |
| Лабораторное занятие №7-8 | №7-8.Использование регулярных выражений |
| Лабораторное занятие №9-10 | №9-10.Операции со списками. |
| Лабораторное занятие №11 | №11.Использование структурных шаблонов. |
| Лабораторное занятие №12-13 | №12-13.Использование поведенческих шаблонов |
| Лабораторное занятие №14 | №14.Установка системы контроля версий |
| Лабораторное занятие №15 | №15.Оптимизация вычислительного алгоритма |
| Лабораторное занятие №16 | №16.Изучение инструментальных средств анализа алгоритмов |
| Лабораторное занятие №17 | №17.Построение событийно-управляемого интерфейса |
| Лабораторное занятие №18 | №18.Создание интерфейсов посредством визуального проектирования |
| Лабораторное занятие №19 | №19.Связывание обработчиков событий с элементами интерфейса |
| Лабораторное занятие №20 | №20.Разработка модуля многооконного интерфейса |
| Лабораторное занятие №21 | №21.Разработка интерфейса пользователя |
| Лабораторное занятие №22 | №22.Создание модуля доступа к БД |

| | |
|--------------------------|--------------------------------|
| Лабораторное занятие №23 | №23.Создание запросов БД |
| Лабораторное занятие №24 | №24.Создание хранимых процедур |

6. САМОСТОЯТЕЛЬНАЯ РАБОТА

| № п/п | Наименование темы (раздела) | Содержание темы (раздела) | Трудоемкость в академических часах |
|-------|---|---|------------------------------------|
| 1 | Тема 1. Жизненный цикл ПО | 1.Анализ требований: изучение различных подходов к анализу требований и их применение в реальных проектах 2.Проектирование: исследование методов и инструментов для проектирования ПО, включая объектно- ориентированное и структурное проектирование | 4 |
| 2 | Тема 2. Структурное программирование | 1.Среда программирования: работа в среде программирования, реализация алгоритмов в виде программ на конкретном языке программирования и оформление кода программы в соответствии со стандартом кодирования 2.Объектно- ориентированная модель программирования: изучение основных принципов объектно- ориентированного программирования на примере алгоритмического языка, понятия классов и объектов, их свойств и методов, инкапсуляции и полиморфизма, наследования и переопределения | 4 |
| 3 | Тема 3. Объектно-ориентированное программирование | 1.Наследование и полиморфизм: использование базовых классов и переопределение методов 2.Работа с файлами и потоками ввода-вывода 3.Примеры реализации практических задач с использованием ООП | 6 |
| 4 | Тема 4. Паттерны проектирования | 1.Фасад (Facade pattern): изучение структуры и принципов работы паттерна, его применение в различных контекстах и реализация на примере конкретного проекта 2.Посетитель (Visitor pattern): изучение структуры и принципов работы паттерна, его применение в различных контекстах и реализация на примере проекта | 4 |
| 5 | Тема5. Событийно-управляемое программирование | 1.Создание и регистрация событий: определение типов событий и их обработка 2.Тестирование событийно-управляемого кода: проверка корректности обработки | 6 |

| | | | |
|---|---|--|---|
| | | событий и взаимодействия объектов | |
| 6 | Тема 6. Оптимизация и рефакторинг кода | 1.Подход для проведения рефакторинга «Выделение функции» в инструменте Klocwork Insight 2.Проект системы рефакторинга, ориентированной на программы обработки разреженных матриц | 4 |
| 7 | Тема 7. Разработка пользовательского интерфейса | 1.Типы пользовательского интерфейса: текстовые, жестовые, нейронные и графические интерфейсы 2.Тактильный, жестовый и нейронный пользовательский интерфейс 3.Этапы UX- и UI- дизайна для сайтов и мобильных приложений | 6 |
| 8 | Тема 8. Службы доступа к данным | 1.Разработка программного модуля для оптимизации выбранной системы доступа к данным 2.Практическая ценность использования средств тестирования для проверки работоспособности систем доступа к данным | 4 |

7. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Вопросы к зачету:

1. Базовые стратегии разработки программных средств и систем. Достоинства. Недостатки. Выбор стратегии в зависимости от характеристик проекта.
2. Введение в технологии разработки программных средств. История развития технологии разработки программного обеспечения. Определения технологии разработки программного обеспечения.
3. Выбор модели жизненного цикла на основе характеристик требований, на основе характеристик команды разработчиков, на основе характеристик коллектива пользователей, на основе характеристик типа проектов и рисков.
4. Выбор модели жизненного цикла на основе характеристик требований, на основе характеристик команды разработчиков, на основе характеристик коллектива пользователей, на основе характеристик типа проектов и рисков.
5. Жизненный цикл программных средств. Стандарт ISO/ IEC 12207:2008 ЖЦ. Основные процессы жизненного цикла программного обеспечения.
6. Жизненный цикл программных средств. Стандарт ISO/ IEC 12207:2008 ЖЦ. Вспомогательные процессы жизненного цикла программного обеспечения.
7. Жизненный цикл программных средств. Стандарт ISO/ IEC 12207:2008 ЖЦ. Организационные процессы жизненного цикла программного обеспечения.
8. Инкрементная модель с уточнением требований на начальных этапах разработки. Инкрементная модель экстремального программирования. Структура. Достоинства. Недостатки. Область применения.
9. Инкрементная стратегия разработки программных средств и систем. Достоинства. Недостатки. Область применения.
10. Каскадная стратегия разработки программных средств и систем. Достоинства. Недостатки. Область применения.
11. Классификация проектов по разработке программных средств и систем. Критерии классификации проектов. Характеристики требований к проекту. Характеристики пользователей (заказчиков).
12. Классификация проектов по разработке программных средств и систем. Критерии

классификации проектов. Характеристики команды разработчиков. Характеристики типов проектов и рисков.

13. Методология функционального моделирования IDEF0. Общие сведения о методологии SADT. Основные понятия IDEF0-модели.

14. Методы нисходящего проектирования. Назначение. Пошаговое уточнение. Проектирование программных средств с помощью псевдокода и управляющих конструкций структурного программирования. Анализ сообщений.

Вопросы к экзамену:

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

2. Дайте характеристику языку ассемблера, определение, преимущества.

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

4. Для чего используются трансляторы, компиляторы, интерпретаторы.

5. Интегрированная среда разработки приложений.

6. Классы в Python. Конструктор класса.

7. Классы в Python. Наследование.

8. Классы в Python. Перегрузка операторов.

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

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

11. Литералы чисел и строк в Python

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

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

14. Область действия переменных и других идентификаторов.

15. Объектно-ориентированное программирование. Компоненты.

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

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

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

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

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

21. Перечислите и опишите этапы проектирования и выполнения программ

22. Перечислите и охарактеризуйте средства аппаратной поддержки функций ОС.

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

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

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

26. Понятие процесса и потока: раскройте и охарактеризуйте.

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

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

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

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

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

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

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

34. Расскажите про компилятор ассемблер - программ, редактор связей (загрузчик).
35. Расскажите про обработку прерываний. Контекст процесса (потока). Особенности операционных систем реального времени.
36. Рекурсия. Механизм рекурсии: действия на рекурсивном спуске и рекурсивном возврате. Понятие о глубине рекурсии. Примеры.
37. Символьный тип данных. Строковый тип данных. Строковые переменные и константы. Двойственный характер строк.
38. Сортировка массивов. C#
39. Списки в Python. List comprehensions: генерация списков.
40. Списки в Python. Методы списков и операции со списками
41. Срезы списков. Присваивание в срез. Проблема копирования списка
42. Строки в Python. Методы find, count, replace, strip, upper, lower и другие.
43. Строки в Python. Срезы с двумя и тремя параметрами
44. Структура программы, элементы языка (алфавит). Понятие типа данных. C#
45. Структурированные типы данных. Линейные массивы. Примеры задач. Двумерные массивы. Примеры задач. C#
46. Условный оператор и каскадная условная конструкция в Python
47. Условный оператор. Оператор множественного ветвления (выбора). C#
48. Файловые типы. Общие процедуры для работы с файлами. Компонентные (типизированные) файлы. C#
49. Функции, ориентированные на работу со строками. Примеры.
50. Функции. Их отличие от процедур. Способ обращения к функции. Примеры.
51. Функции. Организация и вызов. Примеры. C#
52. Цикл for и его особенности в Python. Функция range()
53. Циклы: с параметром. Связь с другими циклами. C#
54. Циклы: с предусловием, с постусловием. Связь с другими циклами. C#
55. Языки программирования. Алгоритмические языки (алфавит, синтаксис, семантика). Способы описания синтаксиса (язык металингвистических формул, синтаксические диаграммы)

| Результаты (освоенные профессиональные компетенции) | Формы и методы контроля и оценки |
|---|--|
| ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием. | <p>Форма контроля: Текущий контроль в форме проверки лабораторных и практических заданий.</p> <p>Методы контроля: Наблюдение за действием обучающихся при выполнении практических заданий.</p> <p>Оценка: – лабораторно-практических работ; – экзамен.</p> |
| ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием. | <p>Форма контроля: Текущий контроль в форме проверки лабораторных и практических заданий.</p> <p>Методы контроля: Оценка: – зачеты по практическим и лабораторным работам; – тесты; – защита лабораторных работ.</p> |
| ПК 1.3. Выполнять отладку программных | Форма контроля: |

| | |
|---|--|
| модулей с использованием специализированных программных средств | Текущий контроль в форме проверки лабораторных и практических заданий. Методы контроля: Оценка: – зачеты по практическим и лабораторным работам; – тесты; – защита лабораторных работ |
|---|--|

8. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ МДК

а) литература

Основная литература:

1. Трофимов, В. В. Основы алгоритмизации и программирования: учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская; под редакцией В. В. Трофимова. — 4-е изд. — Москва: Издательство Юрайт, 2024. — 119 с. — (Профессиональное образование). — ISBN 978-5-534-17498-4. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539994>
2. Белугина, С. В. Разработка программных модулей программного обеспечения для компьютерных систем. Прикладное программирование: учебное пособие / С. В. Белугина. — Санкт-Петербург: Лань, 2020. — 312 с. — ISBN 978-5-8114-4496-0. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/133920>
3. Гниденко, И. Г. Технология разработки программного обеспечения: учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2024. — 248 с. — (Профессиональное образование). — ISBN 978-5-534-18131-9. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539215>

Дополнительная:

1. Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем: учебник для среднего профессионального образования / Е. А. Черткова. — 3-е изд., испр. и доп. — Москва: Издательство Юрайт, 2024. — 146 с. — (Профессиональное образование). — ISBN 978-5-534-18094-7. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539955>
2. Кривоносова, Н. В. Технология WPF. Разработка модулей программного обеспечения: практикум: учебное пособие / Н. В. Кривоносова. — Санкт-Петербург: СПбГУТ им. М.А. Бонч-Бруевича, 2021. — 132 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/279719>
3. Кувшинов, Д. Р. Основы программирования: учебное пособие для среднего профессионального образования / Д. Р. Кувшинов. — Москва: Издательство Юрайт, 2022. — 105 с. — (Профессиональное образование). — ISBN 978-5-534-07560-1. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/493565>
4. Лебедев, В. М. Программирование на VBA в MS Excel: учебное пособие для среднего профессионального образования / В. М. Лебедев. — 3-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2024. — 312 с. — (Профессиональное образование). — ISBN 978-5-534-17043-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/537931>
5. Подбельский, В. В. Программирование. Базовый курс C#: учебник для среднего профессионального образования / В. В. Подбельский. — Москва: Издательство Юрайт, 2024. — 369 с. — (Профессиональное образование). — ISBN 978-5-534-11467-6.

— Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/542233>

6. Казарин, О. В. Основы информационной безопасности: надежность и безопасность программного обеспечения: учебное пособие для среднего профессионального образования / О. В. Казарин, И. Б. Шубинский. — Москва : Издательство Юрайт, 2024. — 342 с. — (Профессиональное образование). — ISBN 978-5-534-10671-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/542339>

б) программное обеспечение и Интернет-ресурсы

| № | Наименование | Описание |
|---|---------------|--|
| 1 | 7-Zip | Бесплатное распространение по лицензии GNU LGPL http://www.7-zip.org/license.txt . |
| 2 | Google Chrome | Бесплатное распространение по лицензии google chromium http://code.google.com/intl/ru/chromium/terms.html на условиях https://www.google.com/chrome/browser/privacy/eula_text.html . |
| 3 | VirtualBox | Бесплатное распространение по лицензии GNU GPL https://www.virtualbox.org/wiki/GPL |
| 4 | LibreOffice | Бесплатное распространение по лицензии GNU LGPL https://ru.libreoffice.org/about-us/license/ |
| 5 | Python 3 | Бесплатное распространение по лицензии GNU GPL http://www.gnu.org/licenses/old-licenses/gpl-2.0.htm . |

в) профессиональные базы данных и информационные справочные системы

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ МДК

Требования к минимальному материально-техническому обеспечению Реализация программы модуля предполагает наличие учебных кабинетов: Занятия учебной дисциплины проводятся в кабинете информационных технологий. Оснащение кабинета: Специализированная мебель и технические средства обучения, служащие для представления учебной информации большой аудитории: учебная мебель, проектор, экран, ПК.