

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Щёкина Вера Витальевна

Должность: Ректор

Дата подписания: 19.07.2021 08:57:31

Уникальный программный ключ:

a2232a55157e576517a8999f3190892af53989420420336ffbf573a434e57789



**МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное бюджетное образовательное  
учреждение высшего образования**

**«Благовещенский государственный педагогический университет»**

**ПРОГРАММА ПОДГОТОВКИ СПЕЦИАЛИСТОВ  
СРЕДНЕГО ЗВЕНА**

**Рабочая программа дисциплины**

**УТВЕРЖДАЮ**

**И.о. декана физико-математического фа-  
культета ФГБОУ ВО «БГПУ»**

**Т.А. Мерделина**

**«29» декабря 2021 г**

**Рабочая программа учебной дисциплины**

**МДК.02.02 ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА РАЗРАБОТКИ  
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**Программа подготовки специалистов среднего звена по специальности  
09.02.07 Информационные системы и программирование**

**Квалификация выпускника  
Программист**

**Принята на заседании кафедры  
информатики и методики преподавания информатики  
(протокол № 5 от «29» декабря 2021 г.)**

**Благовещенск 2021**

## СОДЕРЖАНИЕ

<b>1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА .....</b>	<b>3</b>
<b>2 ТЕМАТИЧЕСКИЙ ПЛАН И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ .....</b>	<b>4</b>
<b>3 УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ .....</b>	<b>5</b>
<b>4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....</b>	<b>6</b>
<b>5 ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ПО ДИСЦИПЛИНЕ .....</b>	<b>7</b>
<b>6 ЛИСТ ИЗМЕНЕНИЙ И ДОПОЛНЕНИЙ .....</b>	<b>18</b>

## 1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

**1.1 Цель дисциплины:** освоение студентами основного вида деятельности «Осуществление интеграции программных модулей» и соответствующих ей профессиональных компетенций.

### **1.2 Место дисциплины в структуре ООП:**

Учебная дисциплина «Инструментальные средства разработки программного обеспечения» (МДК.02.02) принадлежит к профессиональному циклу и основному виду деятельности «Осуществление интеграции программных модулей» (ПМ.02).

### **1.3 Дисциплина направлена на формирование следующих компетенций:**

- ПК 2.2. Выполнять интеграцию модулей в программное обеспечение.
- ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств.
- ПК 2.5 Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

**1.4 Перечень планируемых результатов обучения.** В результате изучения дисциплины обучающийся должен

#### **уметь:**

- организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов;
- выполнять тестирование интеграции;
- использовать различные транспортные протоколы и стандарты форматирования сообщений;
- использовать инструментальные средства отладки программных продуктов;
- использовать методы для получения кода с заданной функциональностью и степенью качества;

#### **знать:**

- основные подходы к интегрированию программных модулей;
- современные технологии и инструменты интеграции;
- методы и способы идентификации сбоев и ошибок при интеграции приложений;
- основные методы отладки;
- приемы работы с инструментальными средствами тестирования и отладки;
- основы организации инспектирования и верификации;
- встроенные и основные специализированные инструменты анализа качества программных продуктов;

#### **иметь практический опыт в:**

- инспектировании разработанных программных модулей на предмет соответствия стандартам кодирования;
- интегрировании модулей в программное обеспечение;
- отладке программных модулей.

**1.5 Общая трудоемкость** дисциплины «Инструментальные средства разработки программного обеспечения» составляет 88 ч. максимальной учебной нагрузки обучающегося, в том числе: обязательной аудиторной учебной нагрузки обучающегося – 76 часов; самостоятельной работы обучающегося – 10 часов.

Программа предусматривает изучение материала на лекциях и лабораторных. Предусмотрена самостоятельная работа обучающихся по темам и разделам. Программа предусматривает использование в образовательном процессе активных и интерактивных

форм проведения занятий в сочетании с внеаудиторной работой. Проверка знаний осуществляется фронтально, индивидуально.

### 1.6 Объем дисциплины и виды учебной деятельности

Вид учебной работы	Объем часов
<b>Максимальная учебная нагрузка (всего)</b>	<b>88</b>
<b>Обязательная аудиторная учебная нагрузка (всего)</b>	<b>76</b>
в том числе:	
- лекции	40
- лабораторные занятия	36
<b>Самостоятельная работа обучающегося (всего)</b>	<b>10</b>
<b>Консультации</b>	
<b>Промежуточная аттестация: дифференцированный зачет</b>	<b>2</b>

## 2 ТЕМАТИЧЕСКИЙ ПЛАН И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Наименование разделов и тем	Содержание учебного материала, лабораторные работы, самостоятельная работа обучающегося	Объем в часах
<b>Тема 1. Современные технологии и инструменты интеграции</b>	<b>Содержание</b>	
	<i>Лекционные занятия:</i> Понятие репозитория проекта, структура проекта. Виды, цели и уровни интеграции программных модулей. Автоматизация бизнес-процессов. Выбор источников и приемников данных, сопоставление объектов данных. Транспортные протоколы. Стандарты форматирования сообщений. Организация работы команды в системе контроля версий.	16
	<i>Лабораторные занятия:</i> Разработка структуры проекта Разработка модульной структуры проекта (диаграммы модулей) Разработка перечня артефактов и протоколов проекта Настройка работы системы контроля версий (типов импортируемых файлов, путей, фильтров и др. параметров импорта в репозиторий) Разработка и интеграция модулей проекта (командная работа) Отладка отдельных модулей программного проекта Организация обработки исключений	18
	<i>Самостоятельная работа обучающихся:</i> Подготовка сообщения об опыте командной разработки на основе материалов сети интернет Знакомство с кратким руководством по GitHub	6
<b>Тема 2. Инстру-</b>	<b>Содержание</b>	

<b>ментарий тестирования и анализа качества программных средств</b>	<i>Лекционные занятия:</i> Отладка программных продуктов. Инструменты отладки. Отладочные классы. Ручное и автоматизированное тестирование. Методы и средства организации тестирования. Инструментарии анализа качества программных продуктов в среде разработки. Обработка исключительных ситуаций. Методы и способы идентификации сбоев и ошибок. Выявление ошибок системных компонентов.	24
	<i>Лабораторные занятия:</i> Применение отладочных классов в проекте Отладка проекта Инспекция кода модулей проекта Тестирование интерфейса пользователя средствами инструментальной среды разработки Разработка тестовых модулей проекта для тестирования отдельных модулей Выполнение функционального тестирования Тестирование интеграции Документирование результатов тестирования	18
	<i>Самостоятельная работа обучающихся:</i> Подготовка сообщения о библиотеке для тестирования программ	4
<b>Консультации</b>		
<b>Промежуточная аттестация:</b> дифференцированный зачет		<b>2</b>
<b>Всего:</b>		<b>88</b>

### 3 УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

#### 3.1 Требования к минимальному материально-техническому обеспечению

Реализация учебной дисциплины требует наличия компьютерного класса – учебная аудитория для проведения всех видов учебных занятий, групповых и индивидуальных консультаций, текущего контроля, промежуточной аттестации и самостоятельной работы.

Комплект учебной мебели, компьютерные столы, аудиторная доска, компьютеры с установленным лицензионным программным обеспечением, мультимедийный проектор, экспозиционный экран, 11 персональных компьютеров.

Используемое программное обеспечение: Microsoft®WINEDUperDVC AllLng Upgrade/SoftwareAssurancePack Academic OLV 1License LevelE Platform 1Year; Microsoft®OfficeProPlusEducation AllLng License/SoftwareAssurancePack Academic OLV 1License LevelE Platform 1Year; Dr.Web Security Suite; Java Runtime Environment; Calculate Linux.

#### 3.2 Информационное обеспечение обучения

##### Литература

1. Гниденко И. Г. Технологии разработки программного обеспечения : учебное пособие для среднего образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. – Москва: Издательство Юрайт, 2021. – 235 с. – (Профессиональное образование). – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/viewer/tehnologiya-razrabotki-programmnogo-obespecheniya-472502>

2. Григорьев М. В. Проектирование информационных систем : учебное пособие для среднего профессионального образования / М. В. Григорьев, И. И. Григорьева. –

Москва: Издательство Юрайт, 2021. – 318 с. – (Профессиональное образование). – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/viewer/proektirovanie-informacionnyh-sistem-476536>

3. Зуб, А. Т. Управление проектами : учебник и практикум для среднего профессионального образования / А. Т. Зуб. — Москва : Издательство Юрайт, 2021. — 422 с. — (Профессиональное образование). — ISBN 978-5-534-01505-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/471393>

4. Черткова Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для среднего профессионального образования / Е. А. Черткова. – 2-е изд., испр. и доп. – Москва: Издательство Юрайт, 2021. – 147 с. – (Профессиональное образование). – Текст: электронный // Образовательная платформа Юрайт [сайт]. – URL: <https://urait.ru/book/programmnyaya-inzheneriya-vizualnoe-modelirovanie-programmnyh-sistem-473307>

#### Базы данных и информационно-справочные системы

1. Национальная платформа «Открытое образование». – Режим доступа: <https://openedu.ru>

2. Образовательная платформа «Лекториум». – Режим доступа: <https://www.lektorium.tv>

3. Электронный фонд нормативно-технической и нормативно-правовой документации Консорциума «Кодекс». – Режим доступа: <https://docs.cntd.ru>

#### Электронно-библиотечные ресурсы

1. ЭБС «Юрайт». – Режим доступа: <https://urait.ru>

2. Полпред (обзор СМИ). – Режим доступа: <https://polpred.com/news>

## 4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения учебной дисциплины осуществляется преподавателем в процессе проведения лекционных занятий, тестирования, лабораторных занятий, а также выполнения обучающимися лабораторного практикума.

Результаты обучения (освоенные умения, усвоенные знания, практический опыт)	Формы и методы контроля результатов обучения
<p><b>Умения:</b></p> <ul style="list-style-type: none"> <li>– организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов;</li> <li>– выполнять тестирование интеграции;</li> <li>– использовать различные транспортные протоколы и стандарты форматирования сообщений;</li> <li>– использовать инструментальные средства отладки программных продуктов.</li> </ul> <p><b>Знания:</b></p> <ul style="list-style-type: none"> <li>– основные подходы к интегрированию программных модулей;</li> <li>– современные технологии и инструменты интеграции;</li> <li>– методы и способы идентификации сбоев и ошибок при интеграции приложений;</li> <li>– основные методы отладки;</li> <li>– приемы работы с инструментальными</li> </ul>	<p>Лабораторный практикум Тест</p> <p>Интерпретация результатов наблюдений за деятельностью обучающегося в процессе выполнения лабораторных работ</p> <p>Защита отчетов в форме собеседования по лабораторным работам</p>

<p>средствами тестирования и отладки;</p> <ul style="list-style-type: none"> <li>– основы организации инспектирования и верификации;</li> <li>– встроенные и основные специализированные инструменты анализа качества программных продуктов.</li> </ul> <p><b>Иметь практический опыт в:</b></p> <ul style="list-style-type: none"> <li>– инспектировании разработанных программных модулей на предмет соответствия стандартам кодирования;</li> <li>– интегрировании модулей в программное обеспечение;</li> <li>– отладке программных модулей.</li> </ul>	
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

## 5 ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ПО ДИСЦИПЛИНЕ

Формируемая компетенция	Показатели освоения компетенций
<p><b>ПК 2.2. Выполнять интеграцию модулей в программное обеспечение.</b></p>	<p><b>Знать:</b></p> <ul style="list-style-type: none"> <li>• основные подходы к интегрированию программных модулей;</li> <li>• современные технологии и инструменты интеграции;</li> <li>• методы и способы идентификации сбоев и ошибок при интеграции приложений;</li> <li>• основные методы отладки;</li> <li>• приемы работы с инструментальными средствами тестирования и отладки;</li> <li>• основы организации инспектирования и верификации.</li> </ul> <p><b>Уметь:</b></p> <ul style="list-style-type: none"> <li>• организовывать заданную интеграцию модулей в программные средства на базе имеющейся архитектуры и автоматизации бизнес-процессов;</li> <li>• выполнять тестирование интеграции;</li> <li>• использовать различные транспортные протоколы и стандарты форматирования сообщений.</li> </ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"> <li>• интегрированием модулей в программное обеспечение.</li> </ul>

**Задание 1.** Выберите из предложенного списка методы интеграции программных модулей:

- а) на уровне брокеров
- б) на уровне программного кода
- в) на уровне интерфейсов
- г) на уровне сервисов
- д) на уровне программистов

Ответ: а, в, г

**Задание 2.** Какие из перечисленных инструментов системы контроля версий позволяют организовать работу команды над проектом?

- а) версионирование папок
- б) управление правами на репозиторий
- в) автоматический поиск ошибок в коде
- г) управление зависимостями между различными файлами проекта
- д) интеграция с системами планирования задач

Ответ: а, б, д

**Задание 3.** Какие из следующих утверждений являются верными для репозитория проекта в системе контроля версий?

- а) Репозитории используются для хранения кода проекта и другой важной информации, такой как история изменений и метаданные.
- б) Репозитории могут быть локальными или удаленными, и они обычно используются для совместной работы над проектом.
- в) Репозиторий может содержать только один проект, и каждый проект должен иметь свой собственный репозиторий.
- г) Все участники проекта должны иметь доступ к репозиторию для внесения изменений и просмотра истории изменений.

Ответ: а, б, г

**Задание 4.** Установите соответствие между видом тестирования и его целью:

- |                   |                                                                                 |
|-------------------|---------------------------------------------------------------------------------|
| 1. интеграционное | а) тестирование минимально возможного для тестирования компонента               |
| 2. нагрузочное    | б) тестирование интегрированной системы на ее соответствие исходным требованиям |
| 3. системное      | в) поиск проблем в интерфейсах и взаимодействии между компонентами программы    |
| 4. модульное      | г) проверка способности программы работать при очень высоких нагрузках          |

Ответ: 1в, 2г, 3б, 4а

**Задание 5.** Какие функции обычно присутствуют в большинстве систем контроля версий с поддержкой репозитория проектов?

- а) регистрация изменений в файлах.
- б) версионирование файлов.
- в) отслеживание авторства изменений.
- г) поддержка ветвления и слияния изменений.

Ответ: а, б, в, г

**Задание 6.** Установите соответствие между уровнем модели OSI и его функцией:

- |                         |                                              |
|-------------------------|----------------------------------------------|
| 1. прикладной уровень   | а) передача данных между приложениями        |
| 2. транспортный уровень | б) передача бит в среде передачи             |
| 3. сетевой уровень      | в) передача пакетов от источника к приемнику |
| 4. канальный уровень    | г) поддержка сетевых приложений              |



5. физический уровень

д) передача данных между соседними узлами сети

Ответ: 1г, 2а, 3в, 4д, 5б

**Задание 7.** Выберите из предложенного списка уровни интеграции программных модулей.

- а) интеграция схем данных
- б) интеграция моделей данных
- в) интеграция кода
- г) интеграция данных
- д) интеграция папок данных

Ответ: а, б, г

**Задание 8.** Установите соответствие:

- |                                        |                                                                                                          |
|----------------------------------------|----------------------------------------------------------------------------------------------------------|
| 1. монолитно-модульная структура       | а) включает в себя программные модули, располагаемые на нескольких уровнях иерархии                      |
| 2. последовательно-модульная структура | б) включает в себя несколько последовательно передающих друг другу управление программных модулей        |
| 3. модульно-иерархическая структура    | в) модули в структуре связаны между собой таким образом, что не образуют в явном виде ни одной структуры |
| 4. модульно-хаотическая структура      | г) включает в себя большой программный модуль, реализующий основную часть функций                        |

Ответ: 1г, 2б, 3а, 4в

**Задание 9.** Установите соответствие:

- |                                       |                                                                                                              |
|---------------------------------------|--------------------------------------------------------------------------------------------------------------|
| 1. пошаговая интеграция               | а) одновременная интеграция нескольких (новых и измененных) компонентов                                      |
| 2. поэтапная интеграция               | б) программная система – набор компонентов с четко определенным интерфейсом                                  |
| 3. компонентно-ориентированный подход | в) код разрабатывается и тестируется малыми компонентами, которые затем постепенно собираются в единое целое |

Ответ: 1в, 2а, 3б

**Задание 10.** Выберите из перечисленных пунктов виды интеграции программных модулей.

- а) линейная интеграция
- б) пошаговая интеграция
- в) суммарная интеграция
- г) плановая интеграция
- д) поэтапная интеграция

Ответ: б, д

**Задание 11.** Выберите из нижеперечисленного протоколы обмена данными:

- а) REST
- б) ISO
- в) API
- г) SOAP
- д) MSF

Ответ: а, в, г

**Задание 12.** Как называется приложение, выполняющее программу в заданном режиме (например, пошаговом) с целью поиска, обнаружения и локализации ошибок?

Ответ: отладчик

**Задание 13.** Как называется процесс разработки и внедрения программного обеспечения, с помощью которого отдельные компоненты могут быть связаны в единую систему?

Ответ: интеграция

**Задание 14.** Как называется набор способов и правил, по которым одна компьютерная программа обменивается данными с другими?

Ответ: программный интерфейс (API)

**Задание 15.** Как называется хранилище информации, связанной с проектом разработки программного продукта в течение всего его жизненного цикла?

Ответ: репозиторий

**Задание 16.** Как называется выделение и связывание информации об одной и той же сущности реального мира из разных коллекций данных?

Ответ: разрешение сущностей

**Задание 17.** Как называется комбинация различных представлений одной и той же сущности реального мира в единое представление?

Ответ: слияние сущностей

**Задание 18.** Напишите алгоритм интеграции двух простых модулей в программное обеспечение. Алгоритм должен включать шаги по настройке взаимодействия между модулями, обработке возможных ошибок и тестированию интеграции.

Содержание верного ответа (допускаются иные формулировки ответа, не искажающие его смысла):

1. Определение целей и требований к интеграции.
2. Анализ существующих решений.
3. Разработка архитектуры интеграции.
4. Создание API для интеграции.
5. Реализация интеграции.
6. Тестирование интеграции.
7. Внедрение интеграции в программное обеспечение.
8. Обучение пользователей.

**Задание 19.** Напишите техническое задание для интеграции модуля авторизации в веб-приложении. Опишите цели интеграции, требования к функционалу, протоколу взаимодействия и критериям оценки.

Содержание верного ответа (допускаются иные формулировки ответа, не искажающие его смысла):

Техническое задание на интеграцию модуля авторизации в веб-приложение:

Цели интеграции: Обеспечить авторизацию пользователей через социальные сети.

Требования к функционалу:

- Поддержка авторизации через различные социальные сети;
- Возможность регистрации и входа через социальные сети;
- Сохранение информации о пользователе после авторизации;
- Отображение информации о текущем авторизованном пользователе.

Протокол взаимодействия: Использовать OAuth 2.0 для авторизации.

Критерии оценки: Успешная авторизация и регистрация через социальные сети, сохранение и отображение информации о пользователе, отсутствие ошибок при авторизации.

**Задание 20.** Приведите последовательность этапов интеграции программного обеспечения.

Содержание верного ответа (допускаются иные формулировки ответа, не искажающие его смысла):

1. Определяем, какой продукт является источником, какой – приемником.
2. Сопоставляем объекты между источником и приемником.
3. Выбираем протокол для интеграции
4. Проводим постобработку данных (после переноса в одну из сторон)

Формируемая компетенция	Показатели освоения компетенций
<b>ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств.</b>	<b>Знать:</b> <ul style="list-style-type: none"><li>• основные подходы к интегрированию программных модулей;</li><li>• методы и способы идентификации сбоев и ошибок при интеграции приложений;</li><li>• основные методы отладки;</li><li>• приемы работы с инструментальными средствами тестирования и отладки.</li></ul> <b>Уметь:</b> <ul style="list-style-type: none"><li>• использовать инструментальные средства отладки программных продуктов.</li></ul> <b>Владеть:</b> <ul style="list-style-type: none"><li>• отладкой программных модулей;</li><li>• инспектированием разработанных программных модулей на предмет соответствия стандартам кодирования.</li></ul>

**Задание 21.** Какие из перечисленных этапов не относятся к процессу отладки программного обеспечения?

- а) определение ошибки
- б) изоляция ошибки
- в) отсеивание несущественных ошибок
- г) тестирование нового функционала

- д) устранение ошибки
- е) проверка работоспособности исправленной программы
- ж) разработка дополнительных функций
- з) версионирование и автоматизация сборки проекта

Ответ: г, ж, з

**Задание 22.** Что не является методом отладки программного обеспечения?

- а) индукция
- б) обратное прослеживание
- в) трансляция
- г) ручное тестирование
- д) дедукция
- е) умозаключение

Ответ: в, е

**Задание 23.** Установите соответствие между названием и функцией инструмента среды программирования:

- |                 |                                                                                                           |
|-----------------|-----------------------------------------------------------------------------------------------------------|
| 1. редактор     | а) создание и изменение исходных символьных файлов, содержащих программу                                  |
| 2. транслятор   | б) формирование исполняемого файла                                                                        |
| 3. библиотекарь | в) перевод программы с языка программирования на язык машинных кодов                                      |
| 4. компоновщик  | г) поддержка совокупности объектных файлов с подпрограммами, типами данных                                |
| 5. загрузчик    | д) копирование исполняемого файла с диска в память и инициализация компьютера перед выполнением программы |

Ответ: 1а, 2в, 3г, 4б, 5д

**Задание 24.** Установите соответствие между термином и его определением:

- |                 |                                             |
|-----------------|---------------------------------------------|
| 1. отладка      | а) процесс поиска ошибок                    |
| 2. тестирование | б) процесс локализации и исправления ошибок |

Ответ: 1б, 2а

**Задание 25.** Установите соответствие:

- |                      |                                                                                                                  |
|----------------------|------------------------------------------------------------------------------------------------------------------|
| 1. ошибки компиляции | а) ошибки, обнаруженные компоновщиком при объединении модулей программы                                          |
| 2. ошибки компоновки | б) ошибки, фиксируемые компилятором при выполнении синтаксического и частично семантического анализа программы   |
| 3. ошибки выполнения | в) ошибки, обнаруженные операционной системой, аппаратными средствами или пользователем при выполнении программы |

Ответ: 1б, 2а, 3в

**Задание 26.** Установите соответствие:

- |                          |                                                                                                |
|--------------------------|------------------------------------------------------------------------------------------------|
| 1. синтаксическая ошибка | а) происходит во время исполнения программы                                                    |
| 2. ошибка исполнения     | б) происходит, когда программа ведет себя не так, как было задумано                            |
| 3. логическая ошибка     | в) происходит, когда встречается код, который не соответствует правилам языка программирования |

Ответ: 1в, 2а, 3б

**Задание 27.** Как называется процесс поиска и исправления ошибок или неполадок в исходном коде какого-либо программного обеспечения?

Ответ: отладка

**Задание 28.** Как называется процесс выполнения программы с целью обнаружения ошибки?

Ответ: тестирование

**Задание 29.** Приведите обобщенный алгоритм отладки программного обеспечения (ПО).

Содержание верного ответа (допускаются иные формулировки ответа, не искажающие его смысла):

1. Выявление проблемы: определение симптомов проблемы, сбор информации о ПО и системе.
2. Выбор инструмента отладки: выбор подходящего отладчика или другого инструмента для работы с ПО.
3. Установка и настройка инструмента отладки на компьютере.
4. Запуск отладки: запуск инструмента отладки, подключение его к ПО и остановка выполнения ПО в точке возникновения проблемы.
5. Локализация ошибки: исследование состояния ПО в момент остановки, определение возможных причин проблемы на основе данных отладчика и анализа кода ПО.
6. Устранение ошибки: исправление ошибочного кода, тестирование исправленного ПО на различных тестовых данных.
7. Завершение отладки: проверка работоспособности ПО после устранения проблемы, подтверждение отсутствия других ошибок, запись результатов отладки в журнал.

**Задание 30.** Перечислите не менее пяти компонентов, входящих в состав интегрированной среды разработки.

Содержание верного ответа (допускаются иные формулировки ответа, не искажающие его смысла):

1. Редактор исходного кода.
2. Компилятор / интерпретатор.
3. Отладчик.
4. Менеджер проектов.
5. Система управления версиями.
6. Средства автоматизации сборки и тестирования.
7. Интеграция с внешними инструментами.

<b>Формируемая компетенция</b>	<b>Показатели освоения компетенций</b>
--------------------------------	----------------------------------------

<p><b>ПК 2.5 Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.</b></p>	<p><b>Знать:</b></p> <ul style="list-style-type: none"> <li>• основы организации инспектирования и верификации;</li> <li>• встроенные и основные специализированные инструменты анализа качества программных продуктов.</li> </ul> <p><b>Уметь:</b></p> <ul style="list-style-type: none"> <li>• использовать методы для получения кода с заданной функциональностью и степенью качества.</li> </ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"> <li>• инспектированием разработанных программных модулей на предмет соответствия стандартам кодирования.</li> </ul>
-------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Задание 31.** Из перечисленных пунктов выберите модели процесса разработки программного обеспечения.

- а) технологическая
- б) каскадная
- в) корневая
- г) итерационная
- д) инкрементальная

Ответ: б, г, д

**Задание 32.** Какие методы используются для обеспечения соответствия разработанных программных модулей стандартам кодирования?

- а) проверка кода
- б) инжиниринг
- в) рециркуляция
- г) рефакторинг
- д) тестирование

Ответ: а, г, д

**Задание 33.** Какие из перечисленных пунктов могут включать стандарты кодирования программного обеспечения?

- а) стандарты представления исходного текста
- б) соглашения по наименованию для компонентов, переменных, констант
- в) регламентацию использования шрифтов и цветов в интерфейсе программы
- г) используемые языки программирования
- д) требования к длине кода в строках

Ответ: а, б, г

**Задание 34.** Какие из перечисленных пунктов являются функциями систем контроля версий?

- а) соблюдение стандарта кодирования
- б) компиляция программы
- в) отладка кода
- г) контроль изменений
- д) поддержка совместной работы

Ответ: г, д

**Задание 35.** Установите соответствие между системой контроля версий и ее типом:

- |             |                     |
|-------------|---------------------|
| 1 Git       | а) централизованная |
| 2 CVS       | б) распределенная   |
| 3 Perforce  |                     |
| 4 Mercurial |                     |

Ответ: 1б, 2а, 3а, 4б

**Задание 36.** Установите соответствие между процессами и их целями:

- |                |                                                                                                                                                                  |
|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 валидация    | а) достижение гарантии того, что объект соответствует требованиям, реализован без непредусмотренных функций и удовлетворяет проектным спецификациям и стандартам |
| 2 верификация  | б) обнаружение ошибок в разрабатываемой программной системе                                                                                                      |
| 3 тестирование | в) доказательство того, что в результате разработки системы достигнуты те цели, которые планировалось достичь благодаря ее использованию                         |

Ответ: 1в, 2а, 3б

**Задание 37.** Как называется набор правил и соглашений, используемых при написании исходного кода на некотором языке программирования?

Ответ: стандарт кодирования (стандарт оформления кода, соглашение по кодированию)

**Задание 38.** Как называется инструмент, позволяющий управлять изменениями в файлах и отслеживать историю этих изменений?

Ответ: система контроля версий (система управления версиями)

**Задание 39.** Исправьте в приведенной программе допущенные несоответствия стандарту кодирования PEP8.

```
while True :
    n = int (input())
    if n == 0:
        break
    if n % 7 == 0 and n%3 == 0:
        print('Караул!')
    elif n % 7:
        print('опасное')
    else:
        print( n)
```

Содержание верного ответа (допускаются иные формулировки ответа, не искажающие его смысла):

```
while True:
    n = int(input())
    if n == 0:
```

```

    break
if n % 7 == 0 and n % 3 == 0:
    print("Караул!")
elif n % 7:
    print("опасное")
else:
    print(n)

```

**Задание 40.** Исправьте в приведенной программе допущенные несоответствия стандарту кодирования PEP8.

```

a = "
w = 0
h = 0
f = 0
flag = True
while a != 'СТОП' :
    a = input ()
    w = w+1
    if ('кот' in a or 'Кот' in a):
        if flag:
            h = w
            flag = False
            f = f + 1
if h == 0:
    print(f,-1)
else:
    print( f, h)

```

Содержание верного ответа (допускаются иные формулировки ответа, не искажающие его смысла):

```

a = ""
w = 0
h = 0
f = 0
flag = True
while a != "СТОП":
    a = input()
    w = w + 1
    if "кот" in a or "Кот" in a:
        if flag:
            h = w
            flag = False
            f = f + 1
if h == 0:
    print(f, -1)
else:
    print(f, h)

```

**Составитель:** Апалеева А.М., преподаватель кафедры информатики и методики преподавания информатики





## **6 ЛИСТ ИЗМЕНЕНИЙ И ДОПОЛНЕНИЙ**

**Утверждение изменений и дополнений в РПД для реализации в 2023/2024 уч. г.**  
РПД обсуждена и одобрена для реализации в 2023/2024 уч. г. на заседании кафедры информатики и методики преподавания информатики (протокол №9 от 26 июня 2023 г.).