1920

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
филиал федерального государственного бюджетного образовательного учреждения
высшего образования «Кубанский государственный университет»
в г. Геленджике

УТВЕРЖДАЮ
Проректор по работе с филиалами
_____ А.А. Евдокимов
«__» августа 2019 г.

Рабочая программа дисциплины

МДК 03.02 ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

специальность 09.02.03 Программирование в компьютерных системах

2019

Рабочая программа учебной дисциплины **МДК.03.02. Инструментальные средства разработки программного обеспечения** разработана на основе Федерального государственного образовательного стандарта (далее – ФГОС) по специальности среднего профессионального образования (далее СПО) 09.02.03 Программирование в компьютерных системах, утвержденного приказом Минобрнауки РФ от 28.07.2014 №804 (зарегистрирован в Минюсте России 21.08.2014 № 33733)

Дисциплина **МДК.03.02. Инструментальные средства разработки программного обеспечения**

Форма обучения	очная
Учебный год	2019-2020
4 курс	7 семестр
лекции	40 час.
практические занятия	30 час.
самостоятельные занятия	40 час.
форма итогового контроля	экзамен

Составитель: преподаватель _____ Л.Л. Левин канд.техн.наук
подпись

Утверждена на заседании предметной (цикловой) комиссии профессиональных дисциплин специальности Программирование в компьютерных системах
протокол № 1 от « 30» августа 2019 г.

Председатель предметной (цикловой) комиссии профессиональных дисциплин специальности Программирование в компьютерных системах
_____ Л.А. Благова « 30 » августа 2019 г.

Рецензент (-ы):

Системный администратор ЗАО «Геленджикский дельфинарий»	_____ подпись, печать	Т.П. Кривошеенко
Директор ООО "Современные Информационные Технологии" г. Геленджик	_____ подпись, печать	А.В. Сметанин

ЛИСТ

согласования рабочей учебной программы по дисциплине

МДК.03.02. Инструментальные средства разработки программного обеспечения

Специальность среднего профессионального образования:
09.02.03 Программирование в компьютерных системах

СОГЛАСОВАНО:

Зам. директора по УР филиала _____ Т. А. Резуненко
« 30 » августа 2019 г.

Заведующая сектором библиотеки _____ Л. Г. Соколова
« 30 » августа 2019 г.

Инженер-электроник (программно-информационное обеспечение образовательной программы) _____ А. В. Сметанин
« 30 » августа 2019 г.

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ 5
1.1 Область применения программы 5
1.2. Место учебной дисциплины в структуре программы подготовки специалистов среднего звена: 5
1.3. Цели и задачи учебной дисциплины – требования к результатам освоения дисциплины: 5
1.4. Перечень планируемых результатов обучения по дисциплине (перечень формируемых компетенций) 6
2.СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ 10
2.1. Объем учебной дисциплины и виды учебной работы 10
2.2. Структура дисциплины: 10
2.3. Тематический план и содержание учебной дисциплины 11
2.4. Содержание разделов дисциплины 14
2.4.1. Занятия лекционного типа 14
2.4.2. Занятия семинарского типа 14
2.4.3. Практические занятия (Лабораторные занятия) 15
2.4.4. Содержание самостоятельной работы (Примерная тематика рефератов) 15
2.4.5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине 16
3. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ 19
3.1.Образовательные технологии при проведении лекций 19
3.2.Образовательные технологии при проведении практических занятий 19
4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ 21
4.1. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине 21
4.2. Перечень необходимого программного обеспечения 21
5. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ 23
5.1. Основная литература 23
5.2. Дополнительная литература 23
5.3. Периодические издания 23
5.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины 24
6. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ 25
7. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ КОНТРОЛЯ УСПЕВАЕМОСТИ 26
7.1. Паспорт фонда оценочных средств 26
7.2. Критерии оценки знаний 26
7.3. Оценочные средств для проведения для текущей аттестации 27
7.4. Оценочные средств для проведения промежуточной аттестации 30
7.4.1. Примерные вопросы для проведения промежуточной аттестации 30
7.4.2. Примерные задачи для проведения промежуточной аттестации 31
8. ДОПОЛНИТЕЛЬНОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ 33

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ МДК.03.02. Инструментальные средства разработки программного обеспечения

1.1. Область применения программы

Рабочая программа учебной дисциплины **МДК.03.02. Инструментальные средства разработки программного обеспечения** является частью программы подготовки специалистов среднего звена в соответствии с ФГОС СПО по специальности 09.02.03 Программирование в компьютерных системах.

1.2. Место учебной дисциплины в структуре программы подготовки специалистов среднего звена:

Дисциплина **МДК.03.02. Инструментальные средства разработки программного обеспечения** входит в профессиональный модуль ПМ.03. «Участие в интеграции программных модулей» основной профессиональной образовательной программы.

Предшествующие дисциплины: Основы программирование, Операционные системы и среды, Прикладное программирование, Системное программирование.

Последующие дисциплины: производственная практика.

1.3. Цели и задачи учебной дисциплины – требования к результатам освоения .

Целью изучения дисциплины является овладение основными понятиями инструментальных средств программного обеспечения. Получение студентами представления о различных системах разработки ПО, трансляторов, компиляторов, интегрированных сред, объектно-ориентированного и визуального программирования, ознакомление их с основными понятиями, принципами, методологией, методиками применения для создания программных продуктов.

Задачи заключаются в изучении следующего:

- Программное обеспечение и его классификация. Понятия прикладного, сервисного и инструментального программного обеспечения.
- Инструментальные программные средства общего и специального назначения.
- Программные и аппаратные инструменты разработки программных средств.
- Инструментальные среды разработки и сопровождения программных средств. Инструментальные среды программирования.
- Понятие компьютерной технологии разработки программных средств и ее рабочие места.

В результате изучения профессионального модуля обучающийся должен:

иметь практический опыт:

участия в выработке требований к программному обеспечению;

участия в проектировании программного обеспечения с использованием специализированных программных пакетов;

уметь:

владеть основными методологиями процессов разработки программного обеспечения;

использовать методы для получения кода с заданной функциональностью и степенью качества;

знать:

модели процесса разработки программного обеспечения;

основные принципы процесса разработки программного обеспечения;
основные подходы к интегрированию программных модулей;
основные методы и средства эффективной разработки;
основы верификации и аттестации программного обеспечения;
концепции и реализации программных процессов;
принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения;
методы организации работы в коллективах разработчиков программного обеспечения;
основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерений характеристик и параметров программ, программных систем и комплексов;
стандарты качества программного обеспечения;
методы и средства разработки программной документации.

Максимальная учебная нагрузка обучающегося 11**0** часов, в том числе:
- обязательная аудиторная учебная нагрузка обучающегося **70** часа;
- самостоятельная работа обучающегося **40** часа.

1.4. Перечень планируемых результатов обучения по дисциплине **МДК.03.02. Инструментальные средства разработки программного обеспечения** (перечень формируемых компетенций)

Учащийся должен обладать **общими** и **профессиональными компетенциями**, включающими в себя способности:

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 3.1. Анализировать проектную и техническую документацию на уровне

взаимодействия компонент программного обеспечения.

ПК 3.2. Выполнять интеграцию модулей в программную систему.

ПК 3.3. Выполнять отладку программного продукта с использованием специализированных программных средств.

ПК 3.4. Осуществлять разработку тестовых наборов и тестовых сценариев.

ПК 3.5. Производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования.

ПК 3.6. Разрабатывать технологическую документацию.

Индекс компетенции	Содержание компетенции (или её части)	В результате изучения учебной дисциплины обучающиеся должны		
		знать	уметь	иметь практический опыт
ОК 1.	Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.	сущность и социальную значимость будущей профессии.	проявлять к будущей профессии устойчивый интерес	-повышение успеваемости по МДК, положительный отзыв руководителя практики. -систематического посещение учебных занятий и практики, консультаций.
ОК 2.	Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.	типовые методы и способы выполнения профессиональных задач.	организовывать собственную деятельность, оценивать эффективность и качество профессиональных задач.	-мотивированного обоснования выбора и применения методов и способов решения профессиональных задач. -точного, правильного и полного выполнения профессиональных задач. -разработки пользовательского интерфейса
ОК 3.	Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	основы нормативной в области разработки и эксплуатации программных продуктов.	принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	-демонстрации способности 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.	перечень профессиональных задач и способы их эффективного решения.	осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.	-обоснования выбора информационных источников для решения профессиональных задач. -оперативности поиска и использования необходимой информации для качественного выполнения профессиональных задач и личностного развития. -использования различных источников информации, включая электронные.
ОК 5.	Использовать информационно-коммуникационные технологии в профессиональной деятельности.	современное ПО для поддержки информационно-коммуникационных технологий	использовать информационно-коммуникационные технологии в профессиональной деятельности.	-осуществления операций с использованием общего и специализированного программного обеспечения. -создания отдельных компонент. -выполнения спецификаций компонент
ОК 6.	Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.	этикет делового общения, основы этики и психологии.	работать в коллективе и в команде, эффективно общаться с коллегами, руководством,	-выполнения задания на учебной практике. -коллективного создания кода программного продукта на уровне модуля в соответствии с готовыми спецификациями.

			потребителями.	
ОК 7.	Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.	Основы теории управления и организации производства при разработке программных продуктов	брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.	-самоанализа и коррекции результатов собственной работы. -выбора специализированных программных средств отладки программных модулей.
ОК 8.	Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.	задачи профессионального и личностного развития.	заниматься самообразованием, осознанно планировать повышение квалификации.	-качественного, своевременного и полного выполнения заданий внеаудиторной самостоятельной работы. -обоснования постановки целей и задач самообразования. -планирования создания кода программного продукта на уровне модуля.
ОК 9.	Ориентироваться в условиях частой смены технологий в профессиональной деятельности.	направления и перспективы развития технологий в области разработки и эксплуатации ПО.	ориентироваться в условиях частой смены технологий в профессиональной деятельности.	-анализа инноваций в области профессиональной деятельности; -отслеживания динамики развития языков программирования и средств его автоматизации.
ПК 3.1.	Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения.	*модели процесса разработки программного обеспечения; *основные принципы процесса разработки программного обеспечения;	* владеть основными методологиями процессов разработки программного обеспечения	* участия в выработке требований к программному обеспечению;
ПК 3.2.	Выполнять интеграцию модулей в программную систему.	*основные подходы к интегрированию программных модулей; *основные методы и средства эффективной разработки;	* использовать методы для получения кода с заданной функциональностью и степенью качества	* участия в проектировании программного обеспечения с использованием специализированных программных пакетов;
ПК 3.3.	Выполнять отладку программного продукта с использованием специализированных программных средств	*основы верификации и аттестации программного обеспечения; *концепции и реализации программных процессов;	* владеть основными методологиями процессов разработки программного обеспечения	* участия в выработке требований к программному обеспечению;
ПК 3.4.	Осуществлять разработку тестовых наборов и тестовых сценариев.	*принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения;	* использовать методы для получения кода с заданной функциональностью и степенью качества	* участия в проектировании программного обеспечения с использованием специализированных программных пакетов;

		*методы организации работы в коллективах разработчиков программного обеспечения;		
ПК 3.5.	Производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования	*основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерений характеристик и параметров программ, программных систем и комплексов;.	* владеть основными методологиями процессов разработки программного обеспечения	* участия в выработке требований к программному обеспечению;
ПК 3.6.	Разрабатывать технологическую документацию.	*стандарты качества программного обеспечения; *методы и средства разработки программной документации.	* использовать методы для получения кода с заданной функциональностью и степенью качества	* участия в проектировании программного обеспечения с использованием специализированных программных пакетов;

2.СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Всего часов
Учебная нагрузка (всего)	**110**
Аудиторные занятия (всего)	**70**
В том числе:	
занятия лекционного типа	**40**
практические занятия (практикумы)	**30**
лабораторные занятия	
Самостоятельная работа (всего)	**40**
в том числе:	
Самостоятельная внеаудиторная работа в виде домашних практических заданий, индивидуальных заданий, самостоятельного подбора и изучения дополнительного теоретического материала	**40**
Вид промежуточной аттестации (экзамен)	Экзамен
Общая трудоемкость час	**110**

2.2. Структура дисциплины:

Наименование разделов и тем		**Количество аудиторных часов**		**Самостоятельная работа студента (час)**
	Всего	**Теоретическое обучение**	**Практические и лабораторные занятия**	
Тема 1. Инструментальные системы технологии программирования	**11**	**4**	**3**	**4**
Тема 2. Структура проекта Delphi	**11**	**4**	**3**	**4**
Тема 3. Средства тестирования и отладки среды Delphi	**11**	**4**	**3**	**4**
Тема 4. Особенности объектно – ориентированного программирования в Delphi	**11**	**4**	**3**	**4**
Тема 5. Создание собственных компонент, пользовательских библиотек процедур и функций в Delphi	**11**	**4**	**3**	**4**
Тема 6. Создание DDL библиотек в Delphi	**11**	**4**	**3**	**4**
Тема 7. Создание многопоточных приложений в Delphi	**11**	**4**	**3**	**4**
Тема 8. Создание приложений баз данных в среде Delphi	**11**	**4**	**3**	**4**
Тема 9. Создание справочной системы	**11**	**4**	**3**	**4**
Тема 10. Средства создания установочного диска.	**11**	**4**	**3**	**4**
Всего по дисциплине	**110**	**40**	**30**	**40**

2.3. Тематический план и содержание учебной дисциплины

МДК.03.02. Инструментальные средства разработки программного обеспечения

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
1	**2**	**3**	**4**
Тема 1. Инструментальные системы технологии программирования .	Содержание учебного материала	***11***	
	Лекции. Инструменты и среды разработки и сопровождения программных средств. Понятие компьютерной технологии разработки программных средств и рабочие места. Понятие инструментальных систем технологии программирования. Характеристики компьютерной поддержки.	4	*2*
	Практические занятия. Тема: Компоненты инструментальных систем технологии программирования.	**3**	2
	Самостоятельная работа обучающихся: Проработка конспектов занятий, проверка усвоенного материала по теме.	*4*	2
Тема 2. Структура проекта Delphi	Содержание учебного материала	***11***	
	Лекции. Визуальное проектирование в среде Delphi	4	*3*
	Практические занятия. Тема: Интерфейс среды разработки Delphi	***3***	2
	Самостоятельная работа обучающихся: 1. Директивы компилятора, подробная детализация и использование в приложениях; 2. Работа с указателями, процедурные и вариантные типы данных	4	2
Тема 3. Средства тестирования и отладки среды Delphi	Содержание учебного материала	***11***	
	Лекции. Классификация ошибок. Обработка исключительных ситуаций. Классы исключений. Тестирование и отладка программы.	4	2
	Практические занятия. Тема: Тестирование и отладка программ в среде Delphi.	**3**	2
	Самостоятельная работа обучающихся: Создание и использование собственных модулей; Создание пользовательских классов – методика, апробация, применение.	4	2
Тема 4. Особенности объектно – ориентированного программирования в Delphi	Содержание учебного материала	**11**	
	Лекции. Классы и объекты Object Pascal. Основные свойства ООП в Delphi.	4	2
	Практические занятия. Обработка исключительных ситуаций в Delphi.	**3**	2
	Самостоятельная работа обучающихся: Профессиональная настройка среды Delphi; Работа с классами в среде визуального программирования.	*4*	2
Тема 5. Создание	Содержание учебного материала	***11***	
	Лекции. Основные этапы создания компонент. Выбор базового	4	2

собственных компонент, пользовательских библиотек процедур и функций в Delphi	класса. Создание модуля компонента. Тестирование и установка компонента. Понятие и виды подпрограмм. Особенности описания и создания процедур и функций. Создание и использование библиотек и подпрограмм..		
	Практические занятия. Тема: Проектирование многооконных приложений в Delphi.	***3***	2
	Самостоятельная работа обучающихся: Обработка исключительных ситуаций, защита ресурсов и генерации исключений; Пакеты компонентов в среде Delphi;	4	2
Тема 6. Создание DDL библиотек в Delphi.	Содержание учебного материала	***11***	
	Лекции. Понятие и особенности DDL – библиотек. Структура проекта DDL. Вызов подпрограммы DDL.	4	2
	Практические занятия. Тема: Создание и тестирование DDL – библиотек средствами Delphi.	**3**	2
	Самостоятельная работа обучающихся: Создание SDI и MDI приложений; Технология доступа к данным ADO.	4	2
Тема 7. Создание многопоточных приложений в Delphi	Содержание учебного материала	**11**	
	Лекции. Понятие потоков и процессов. Средства синхронизации потоков. Создание модулей – потоков в Delphi. Свойства и методы потоков.	4	2
	Практические занятия. Тема: Разработка многопоточного приложения в Delphi	**3**	2
	Самостоятельная работа обучающихся: Освоение редко используемых компонент Палитры компонент среды Delphi; Профессиональная работа с наборами данных.	4	2
Тема 8. Создание приложений баз данных в среде Delphi	Содержание учебного материала	***11***	
	Лекции. Компоненты, используемые для работы с БД. Подключение и отображение наборов данных. Визуальные компоненты отображения данных.	4	2
	Практические занятия. Тема: Средства создания структуры базы данных в среде Delphi.	**3**	2
	Самостоятельная работа обучающихся: Освоение инструментальных средств по работе с базами данных; Работа со связанными таблицами в рамках баз данных; Работа с сокетами; Введение в создание компонентов;	4	2
Тема 9. Создание справочной системы	Содержание учебного материала	***11***	
	Лекции. Комплекс мероприятий, направленных на сопровождение ПО.	4	2
	Практические занятия. Тема: Создание и настройка справочной системы приложения средствами в Delphi.	**3**	2
	Самостоятельная работа обучающихся: Формирование отчетов в среде Delphi; Применение языка SQL для создания запросов к базе данных; Перенос приложений из среды Windows в среду Linux.	*4*	2
	Содержание учебного материала	***4***	

Тема 10. Средства создания установочного диска.	**Лекции.** Проект и структура программы установки. Выбор устанавливаемых компонентов и конфигурирование системы пользователя. Создание образа установочного диска.		*2*
	Практические занятия. Тема: Создание установочного диска	**3**	2
	Самостоятельная работа обучающихся: Использование XML файлов при работе в среде Delphi; Разработка Web – приложений в среде Delphi; Перенос приложений баз данных и Web – приложений; Компоненты Intra Web, создание приложений с помощью Intra Web.	*4*	2
Итого	Лекции	***40***	
	Практические занятия.	**30**	
	Самостоятельная работа обучающихся	40	
	Всего	**110**	

Для характеристики уровня освоения учебного материала используются следующие обозначения: 1. – ознакомительный (узнавание ранее изученных объектов, свойств); 2. – репродуктивный (выполнение деятельности по образцу, инструкции или под руководством) 3. – продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач)

2.4. Содержание разделов дисциплины

2.4.1. Занятия лекционного типа

№ темы	Наименование темы	Содержание темы	Форма текущего контроля
1	2	3	4
1	Инструментальные системы технологии программирования	Инструменты и среды разработки и сопровождения программных средств. Понятие компьютерной технологии разработки программных средств и рабочие места. Понятие инструментальных систем технологии программирования. Характеристики компьютерной поддержки.	Т, У
2	Структура проекта Delphi	Визуальное проектирование в среде Delphi	Т, У
3	Средства тестирования и отладки среды Delphi	Классификация ошибок. Обработка исключительных ситуаций. Классы исключений. Тестирование и отладка программы.	Т, У
4	Особенности объектно – ориентированного программирования в Delphi	Классы и объекты Object Pascal. Основные свойства ООП в Delphi.	Т, У
5	Создание собственных компонент, пользовательских библиотек процедур и функций в Delphi	Основные этапы создания компонент. Выбор базового класса. Создание модуля компонента. Тестирование и установка компонента. Понятие и виды подпрограмм. Особенности описания и создания процедур и функций. Создание и использование библиотек и подпрограмм..	Т, У
6	Создание DDL библиотек в Delphi	Понятие и особенности DDL – библиотек. Структура проекта DDL. Вызов подпрограммы DDL.	Т, У
7	Создание многопоточных приложений в Delphi	Понятие потоков и процессов. Средства синхронизации потоков. Создание модулей – потоков в Delphi. Свойства и методы потоков	Т, У
8	Создание приложений баз данных в среде Delphi	Компоненты, используемые для работы с БД. Подключение и отображение наборов данных. Визуальные компоненты отображения данных	Т, У
9	Создание справочной системы	Комплекс мероприятий, направленных на сопровождение ПО.	Т, У
10	Средства создания установочного диска	Проект и структура программы установки. Выбор устанавливаемых компонентов и конфигурирование системы пользователя. Создание образа установочного диска.	Т, У
Примечание: Т – тестирование, У – устный опрос,			

2.4.2. Занятия семинарского типа

 не предусмотрены

2.4.3. Практические занятия (Лабораторные занятия)

№ темы	Наименование темы	Содержание темы	Форма текущего контроля
1	2	3	4
1	Инструментальные системы технологии программирования	Компоненты инструментальных систем технологии программирования.	Т, У
2	Структура проекта Delphi	Интерфейс среды разработки Delphi	Т, У
3	Средства тестирования и отладки среды Delphi	Тестирование и отладка программ в среде Delphi.	Т, У
4	Особенности объектно – ориентированного программирования в Delphi	Обработка исключительных ситуаций в Delphi.	Т, У
5	Создание собственных компонент, пользовательских библиотек процедур и функций в Delphi	Проектирование многооконных приложений в Delphi.	Т, У
6	Создание DDL библиотек в Delphi	Создание и тестирование DDL – библиотек средствами Delphi.	Т, У
7	Создание многопоточных приложений в Delphi	Разработка многопоточного приложения в Delphi	Т, У
8	Создание приложений баз данных в среде Delphi	Средства создания структуры базы данных в среде Delphi.	Т, У
9	Создание справочной системы	Создание и настройка справочной системы приложения средствами в Delphi., направленных на сопровождение ПО.	Т, У
10	Средства создания установочного диска	Создание установочного диска	Т, У
Примечание: Т – тестирование, У – устный опрос,			

2.4.4. Содержание самостоятельной работы

Придумать алгоритм, написать программу для игры и отладить её:

1. Игра «Сбей самолет». По экрану летят вражеские самолеты. Цель — их сбить. В нижней строке экрана находится пусковая установка, которую можно перемещать по строке вперед и назад.
2. Игра «100 спичек». Из кучки, первоначально содержащей 100 спичек, двое играющих поочередно берут по несколько спичек: не менее одной и не более десяти, Проигрывает взявший последнюю спичку.
3. Игра «Крестики - нолики». Написать программу.
4. Игра «Расстановка девяти чисел». В квадрате размером 3 х 3 клетки расставить числа 1, 2, 3,..., 9 так, чтобы суммы чисел, стоящих в каждом вертикальном ряду, в каждом горизонтальном ряду, а также на любой диагонали были равны.
5. Игра «Угадай число». Программа с помощью датчика случайных чисел выбирает число в диапазоне от 0 до 9. Угадать это число за три попытки. После каждой попытки сообщает, больше или меньше названное задуманного.
6. Игра «Баше». В игре участвуют двое. Условия игры: имеются N предметов. Соперники ходят по очереди. За каждый ход игрок может взять 1, 2, ..., К предметов. Проигрывает тот, кто вынужден взять последний предмет.
7. Игра «Скачки». В игре участвуют 10 наездников; за каждый тур игры каждый из них продвигается вперед на расстояние от 1 до 5 км случайным образом. Длина дистанции — 50 км. Всего 5 заездов, победителю заезда начисляется 5 очков.

Победителем считается наездник, набравший наибольшее количество очков во всех заездах. Перед началом заездов участник игры выбирает номер наездника.

8. Игра «Гонка с выбыванием». В мотокроссе участвуют 15 спортсменов, участники преодолевают 14 кругов. После каждого круга участник, занимающий последнее место, снимается с соревнований. Перед началом соревнований участники игры разыгрывают номера. Количество участников не более 15.

9. Игра «Морской бой». Игра происходит между 2 участниками на поле 10 х 10 клеток. Участники расставляют 10 одноклеточных катеров. При попадании — катер уничтожен, участник делает еще ход. Ход в клетку, соседнюю с уничтоженным кораблем, недействителен и повторяется. Победитель — первый, уничтоживший катера соперника.

2.4.5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Самостоятельная работа студентов является важнейшей формой учебно-познавательного процесса.

Основная цель самостоятельной работы студента при изучении дисциплины – закрепить теоретические знания, полученные в ход лекционных занятий, а также сформировать практические навыки подготовки в области Инструментальных средств разработки программного обеспечения.

Самостоятельная работа студента в процессе освоения дисциплины включает:

-изучение основной и дополнительной литературы по курсу;

-самостоятельное изучение некоторых вопросов (конспектирование);

-работу с электронными учебными ресурсами;

-изучение материалов периодической печати, интернет ресурсов;

-подготовку к тестированию;

-подготовку к практическим занятиям,

-самостоятельное выполнение домашних заданий.

Для помощи в самостоятельной работе рекомендуется применять электронный учебник (учебное пособие) **Программирование.СНМ**, разработанное Левиным Л.Л.

На самостоятельную работу студентов отводится 40 часов учебного времени.

№	Наименование темы, вида СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	Инструментальные системы технологии программирования	1. Голицына О.Л. Программное обеспечение: учебное пособие / Голицына О.Л., Партыка Т.Л., Попов И.И., - 4-е изд., перераб. и доп. - М.: Форум: ИНФРА-М, 2017. - 448 с. 2. *Советов, Б. Я.* Информационные технологии [Электронный ресурс]: учебник для СПО / Б. Я. Советов, В. В. Цехановский. — 6-е изд., перераб. и доп. — М.: Издательство Юрайт, 2017. — 261 с. – URL: https://www.biblio-online.ru/book/BBC6F436-97B4-4DCB-829E-1DF182A8B1A4 3. Инструментальные средства информационных систем [Электронный ресурс]: учебное пособие / Л.В. Абрамова. – Архангельск: САФУ, 2013. – 118 с. – URL: http://biblioclub.ru/index.php?page=book&id=436131 4. *Черпаков, И. В.* Основы программирования [Электронный ресурс]: учебник и практикум для СПО / И. В. Черпаков. — М.: Издательство Юрайт, 2017. — 219 с. – URL: https://www.biblio-online.ru/book/F79BE55A-C6F1-439D-9ED5-0D78A50B403F
2	Структура проекта Delphi	
3	Средства тестирования и отладки среды Delphi	
4	Особенности объектно – ориентированного программирования в Delphi	
5	Создание собственных компонент, пользовательских библиотек процедур и функций в Delphi	
6	Создание DDL библиотек в Delphi	
7	Создание многопоточных приложений в Delphi	
8	Создание приложений баз данных в среде Delphi	
9	Создание справочной системы	

10	Средства создания установочного диска	5. Вичугова, А.А. Инструментальные средства информационных систем [Электронный ресурс]: учебное пособие / А.А. Вичугова. - Томск: Издательство Томского политехнического университета, 2015. - 136 с. - URL: http://biblioclub.ru/index.php?page=book_red&id=442814&sr=1

Кроме перечисленных источников студент может воспользоваться поисковыми системами сети Интернет по теме самостоятельной работы.

Для освоения данной дисциплины и выполнения предусмотренных учебной программой курса заданий по самостоятельной работе студент может использовать следующее учебно-методическое обеспечение:

-обучающие видеофильмы и программы по тематике решаемых задач из **Видеотеки программирования** филиала (225 единиц);

-программу компьютерного обучения и контроля “**ЭкзамL**”;

-электронный учебник по прикладному программированию;

-методические рекомендации преподавателя к лекционному материалу;

-методические рекомендации преподавателя к практическим занятиям;

-методические рекомендации преподавателя к выполнению самостоятельных домашних заданий.

Началом организации любой самостоятельной работы должно быть привитие навыков и умений грамотной работы с учебной и научной литературой. Этот процесс, в первую очередь, связан с нахождением необходимой для успешного овладения учебным материалом литературой. Студент должен уметь пользоваться фондами библиотек и справочно-библиографическими изданиями.

Студенты для полноценного освоения учебного курса должны составлять конспекты как при прослушивании его теоретической (лекционной) части, так и при подготовке к практическим (лабораторным) занятиям. Желательно, чтобы конспекты лекций записывались в логической последовательности изучения курса и содержались в одной тетради.

3. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Для реализации компетентностного подхода предусматривается использование в учебном процессе компьютерных активных и интерактивных форм проведения аудиторных и внеаудиторных занятий с целью формирования и развития профессиональных навыков обучающихся.

В процессе преподавания применяются образовательные технологии развития критического мышления. Обязательны компьютерные практические работы по разделам дисциплины.

В учебном процессе наряду с традиционными образовательными технологиями используются электронные учебники, компьютерное обучение, тестирование, учебные видеофильмы, тематические презентации, интерактивные технологии.

3.1. Образовательные технологии при проведении лекций

№	Тема	Виды применяемых образовательных технологий	Кол-во час
1	2	3	4
1	Инструментальные системы технологии программирования	Компьютерные технологии обучения, активное обучение, электронный учебник, тестирование.	**4***
2	Структура проекта Delphi		**4***
3	Средства тестирования и отладки среды Delphi		**4***
4	Особенности объектно – ориентированного программирования в		**4***
5	Создание собственных компонент, пользовательских библиотек		**4***
6	Создание DDL библиотек в Delphi		**4***
7	Создание многопоточных приложений в Delphi		**4***
8	Создание приложений баз данных в среде Delphi		**4***
9	Создание справочной системы		**4***
10	Средства создания установочного диска		**4***
	Итого по курсу		40
	в том числе интерактивное обучение*		40*

3.2.Образовательные технологии при проведении практических занятий

№	Тема занятия	Кол. час	Виды применяемых образовательных технологий
1	Компоненты инструментальных систем технологии программирования.	**3***	Компьютерные технологии обучения. Электронный учебник. Активное обучение. Дискуссия по теоретическим вопросам. Решение задач индивидуально. Решение задач малыми группами. Разбор решения задач.
2	Интерфейс среды разработки Delphi	**3***	
3	Тестирование и отладка программ в среде Delphi.	**3***	
4	Обработка исключительных ситуаций в Delphi.	**3***	
5	Проектирование многооконных приложений в Delphi.	**3***	
6	Создание и тестирование DDL – библиотек средствами Delphi.	**3***	
7	Разработка многопоточного приложения в Delphi	**3***	
8	Средства создания структуры базы данных в среде Delphi.	**3***	
9	Создание и настройка справочной системы приложения средствами в Delphi., направленных на сопровождение ПО.	**3***	
10	Создание установочного диска	**3***	

	Итого по курсу	30	
	в том числе интерактивное обучение*	30*	

4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ

4.1. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Реализация учебной дисциплины осуществляется в специально оборудованном компьютерном классе.

Оборудование учебного кабинета:

- мультимедийный проектор, экран;
- персональный компьютер, динамики;
- выход в Интернет;
- учебная мебель;
- доска учебная;

Наглядные пособия:

1. Видеофильм Программное прикладное обеспечение.mp4
2. Видеофильм Классификация ПО.mp4
3. 3 видеофильма Программирование игры пинг-понг. mp4
4. 2 видеофильма Проектирование меню приложения. mp4
5. 2 видеофильма Использование компоненты Mediaplayer. mp4
6. 2 видеофильма Создаём Браузер mp4
7. Видеофильм Как работать с папками в Delphi.mp4

Электронные ресурсы:

1. Технология разработки прикладного программного обеспечения https://www.monographies.ru/ru/book/view?id=141
2. Справочник Delphi http://delphimaster.net/ Delphi Master Search Archive
3. Учебник Delphi http://www.delphi-manual.ru/ Уроки Delphi начинающим с нуля
4. Delphi компоненты. Справочник http://www.delphisources.ru/
5. Delphi Форум программистов http://www.programmersforum.ru/index.php
6. Он-лайн справочник. Основы Delphi http://www.delphibasics.ru/

4.2. Перечень необходимого программного обеспечения

1. Lazarus – визуальная среда программирования (в свободном доступе);
2. PascalABC - визуальная среда программирования (в свободном доступе);
3. PascalABC.NET - визуальная среда программирования (в свободном доступе);
4. WEB-среда разработки PascalABC.NET. URL http://wde.pascalabc.net/ (в свободном доступе);
5. Среда разработки ПО PortableDelphi. (в свободном доступе);
6. Разработчик инсталляторов InnoSetup. (в свободном доступе);
7. 7-zip архиватор; (лицензия на англ. http://www.7-zip.org/license.txt)
8. Adobe Acrobat Reader просмотрщик файлов ; (лицензия - https://get.adobe.com/reader/?loc=ru&promoid=KLXME)
9. Adobe Flash Player –графический редактор; (лицензия - https://get.adobe.com/reader/?loc=ru&promoid=KLXME)
10. Apache OpenOffice – офисный пакет; (лицензия - http://www.openoffice.org/license.html)

11. FreeCommander - проводник; (лицензия - https://freecommander.com/ru/%d0%bb%d0%b8%d1%86%d0%b5%d0%bd%d0%b7%d0%b8%d1%8f/)
12. Google Chrome - браузер;(лицензия - https://www.google.ru/chrome/browser/privacy/eula_text.html)
13. LibreOffice – офисный пакет (в свободном доступе);
14. Mozilla Firefox - браузер.(лицензия - https://www.mozilla.org/en-US/MPL/2.0/)
15. nanoCAD версия 5.1 локальная (лицензия - серийный номер: NC50B-45103)
16. ЭкзамL – Система компьютерного тестирования http://Lkub.ru Левин Л.Л. (в свободном доступе);
17. Программный комплекс "Универсальный тест 4.0.0.1" http://www.timk.ru/ (в свободном доступе);

5. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ МДК.03.02. Инструментальные средства разработки программного обеспечения

5.1. Основная литература

1. Советов, Б. Я. Информационные технологии [Электронный ресурс]: учебник для СПО / Б. Я. Советов, В. В. Цехановский. — 6-е изд., перераб. и доп. — М.: Издательство Юрайт, 2018. — 261 с. – URL: https://biblio-online.ru/viewer/BBC6F436-97B4-4DCB-829E-1DF182A8B1A4#page/1
2. Черпаков, И. В. Основы программирования [Электронный ресурс]: учебник и практикум для СПО / И. В. Черпаков. — М.: Издательство Юрайт, 2018. — 219 с. – URL: https://biblio-online.ru/viewer/F79BE55A-C6F1-439D-9ED5-0D78A50B403F#page/1

5.2. Дополнительная литература

1. Голицына О.Л. Программное обеспечение: учебное пособие / Голицына О.Л., Партыка Т.Л., Попов И.И., - 4-е изд., перераб. и доп. - М.: Форум: ИНФРА-М, 2017. - 448 с. 30
2. Лаврищева, Е. М. Технология программирования и программная инженерия [Электронный ресурс]: учебник / Е. М. Лаврищева. — М.: Издательство Юрайт, 2018. — 432 с. - URL: https://biblio-online.ru/viewer/DC92DBFC-36FA-4397-B521-4753ABFE25D9#page/1
3. Зыков, С. В. Программирование. Объектно-ориентированный подход [Электронный ресурс]: учебник и практикум / С. В. Зыков. — М.: Издательство Юрайт, 2018. — 155 с. - URL: https://biblio-online.ru/viewer/343E3BED-A03E-4029-AA21-2ABC53126EFA#page/1

.

5.3 Периодические издания

1. Среднее и профессиональное образование
2. Компьютер Пресс
3. Открытые системы.- URL: http://biblioclub.ru/index.php?page=journal&jid=436083
4. Информатика в школе .- URL: http://dlib.eastview.com/browse/publication/18988
5. Программные продукты и системы.- URL: http://dlib.eastview.com/browse/publication/64086
6. Информатика и образование.- URL: http://dlib.eastview.com/browse/publication/18946
7. Системный администратор.- URL: http://dlib.eastview.com/browse/publication/66751
8. Computerword Россия.- URL: http://dlib.eastview.com/browse/publication/64081
9. Мир ПК.- URL: http://dlib.eastview.com/browse/publication/64067
10. Информационно-управляющие системы.- URL: http://dlib.eastview.com/browse/publication/71235
11. Журнал сетевых решений LAN.- URL: http://dlib.eastview.com/browse/publication/64078
12. .Информатика и образование.- URL: http://dlib.eastview.com/browse/publication/18946
13. Windows IT Pro/ Re.- URL: http://biblioclub.ru/index.php?page=journal&jid=138741
14. Прикладная информатика.- URL: http://elibrary.ru/title_about.asp?id=25599

5.4 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. ЭБС «Университетская библиотека ONLINE». – URL: www.biblioclub.ru
2. ЭБС издательства «Лань». – URL: https://e.lanbook.com
3. ЭБС «Юрайт». – URL: http://www.biblio-online.ru/
4. ЭБС «BOOK.ru» https://www.book.ru
5. ЭБС «ZNANIUM.COM» www.znanium.com
6. Электронный каталог Научной библиотеки КубГУ. – URL:http://212.192.134.46/MegaPro/Catalog/Home/Index
7. Электронная библиотека «Издательского дома «Гребенников» - URL:www.grebennikon.ru
8. Научная электронная библиотека (НЭБ) «eLibrary.ru». - URL:http://www.elibrary.ru
9. Базы данных компании «Ист Вью». - URL:http://dlib.eastview.com
10. Лекториум ТВ». - URL: http://www.lektorium.tv/
11. Национальная электронная библиотека «НЭБ». - URL:http://нэб.рф/
12. КиберЛенинка: научная электронная библиотека. – URL: http://cyberleninka.ru/
13. Единое окно доступа к образовательным ресурсам : федеральная ИС свободного доступа. – URL: http://window.edu.ru.
14. Справочно-правовая система «Консультант Плюс» - URL http://www.consultant.ru

6. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Методические указания по выполнению практических работ

Перечень практических заданий

Задание 1 Разработать функциональную модель декомпозиции учета движения материалов на складе фирмы

Задание 2 Разработать функциональную модель работы информационной системы приемной комиссии института

Задание 3 Разработать функциональную модель декомпозиции работы информационно-справочной службы фирмы

Задание 4 Разработать функциональную модель работы информационной системы городского бюро медико-социальной экспертизы

Задание 5 Разработать функциональную модель декомпозиции работы информационной системы туристической фирмы

Задание 6 Разработать функциональную модель работы офиса продаж оператора сотовой связи

Задание 7 Разработать функциональную модель декомпозиции работы бухгалтерии предприятия

Задание 8 Разработать функциональную модель работы переговорного пункта

Задание 9 Разработать функциональную модель декомпозиции работы регистратуры больницы

Задание 10 Разработать функциональную модель декомпозиции работы отдела кадров предприятия

Задание 11 Разработать функциональную модель работы учебного отдела вуза

Задание 12 Разработать функциональную модель декомпозиции работы деканата СПО

Задание 13 Разработать модель информационной системы страховой компании

Задание 14 Разработать модель информационной системы пункта проката видеофильмов

Задание 15 Разработать модель информационной системы начисления сдельной заработной платы

Задание 16 Разработать модель информационной системы учета транспортных перевозок

Задание 17 Разработать модель информационной системы кассы автостанции

Задание 18 Разработать в модель информационной системы учета заявок клиентов торговой фирмы

7. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ КОНТРОЛЯ УСПЕВАЕМОСТИ

7.1. Паспорт фонда оценочных средств

Код и наименование элемента **знаний,** контролируемые компетенции

31	модели процесса разработки программного обеспечения;	ПК 3.1.
32	основные принципы процесса разработки программного обеспечения;	ПК 3.2.
33	основные подходы к интегрированию программных модулей;	ПК 3.3.
34	основные методы и средства эффективной разработки;	ПК 3.4.
35	основы верификации и аттестации программного обеспечения;	ПК 3.5.
36	концепции и реализации программных процессов;	ПК 3.6.
37	принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения;	ПК 3.4.

З8	методы организации работы в коллективах разработчиков программного обеспечения;	ПК 3.2.
З9	основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерений характеристик и параметров программ, программных систем и комплексов;	ПК 3.1.
З10	стандарты качества программного обеспечения;	ПК 3.6.
З11	методы и средства разработки программной документации.	ПК 3.5.

Код и наименование элемента **умений,** контролируемые компетенции

У1	владеть основными методологиями процессов разработки программного обеспечения;	ПК 3.1.
У2	использовать методы для получения кода с заданной функциональностью и степенью качества;	ПК 3.2.

Код и наименование элемента **практического опыта,** контролируемые компетенции

О1	участия в выработке требований к программному обеспечению;	ПК 3.4.
О2	участия в проектировании программного обеспечения с использованием специализированных программных пакетов;	ПК 3.5.

Для оценки вышеуказанных знаний и умений используются программы, разработанные и отлаженные обучающимся, которые представлены преподавателю. Дополнительный контроль проводится с помощью тематических тестов и собеседований.

7.2. Критерии оценки знаний

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в процессе проведения практических занятий, тестирования, а также выполнения обучающимися индивидуальных самостоятельных заданий.

Тест. Система стандартизированных заданий, позволяющая автоматизировать процедуру измерения уровня знаний и умений обучающегося. Тест оценивается по количеству правильных ответов, по времени выполнения, весу (сложности) заданий (не менее 50%).

Критерии оценки знаний студентов в целом по дисциплине:

«отлично» - выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений;

«хорошо» - выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;

«удовлетворительно» - выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;

«неудовлетворительно» - выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач.

7.3. Оценочные средств для проведения текущей аттестации

В данном разделе приводятся образцы оценочных средств. Полный комплект оценочных средств приводится в Фонде оценочных средств.

Текущий контроль проводится в форме:

-тестирование по теоретическому материалу

-практическая работа – разработка и отладка программы

-защита выполненного задания,

-индивидуальный устный опрос.

Форма аттестации	Знания	Умения	Практический опыт (владение)	Личные качества студента	Примеры оценочных средств
Устный (письменный) опрос по темам	Контроль знаний по закладкам Delphi с встроенными компонентами	Оценка умения применять компоненты для разработки программ	Оценка навыков разработки законченных программ	Оценка способности оперативно и качественно отвечать на поставленные вопросы	Контрольные вопросы по темам прилагаются
Практические работы	Контроль знания основ программирования свойств и событий компонент Delphi	Оценка умения работать с графическими компонентами Delphi	Оценка навыков работы прикладными программными средствами	Оценка способности качественно решать задачи и аргументировать результаты	Темы работ прилагаются
Тестирование	Контроль знаний по определенным темам	Оценка умения различать конкретные понятия алгоритмов	Оценка навыков логического анализа и синтеза при сопоставлении конкретных понятий	Оценка способности оперативно и качественно отвечать на поставленные вопросы	Вопросы прилагаются

Варианты заданий:

1. Создать файл, содержащий данные: ФИО ученика; класс; оценка. Просмотреть записи. Определить среднюю оценку в 9-м классе.
2. Создать файл, содержащий данные: ФИО студента; группа; оценка по информатике. Просмотреть записи. Определить среднюю оценку по информатике в 302 группе.
3. Создать файл, содержащий данные: ФИО работника; цех; стаж работы. Просмотреть записи. Определить средний стаж работы работников 1 - го цеха.
4. Создать файл, содержащий данные: ФИО работника; пол; количество детей. Просмотреть данные из файла. Определить количество женщин, имеющих более двух детей.
5. Создать файл, содержащий данные: ФИО ученика; пол; рост. Просмотреть данные из файла. Определить средний рост девочек.
6. Создать файл, содержащий данные: ФИО ученика; класс; пол. Просмотреть данные из файла. Определить количество мальчиков в 5 - классе.

7. Создать файл, содержащий данные: ФИО студента; группа; оценка по информатике. Просмотреть данные из файла. Определить количество ‘4’ и‘5’ в 301 группе.
8. Создать файл, содержащий следующие данные: ФИО студента; пол; группа. Вывести данные из файла на экран. Подсчитать количество юношей и количество девушек 301 группы. Создать новый файл данных, содержащий сведения о студентах 302 группы (используя существующий файл данных).
9. Создать файл, содержащий следующие данные: ФИО студента; год рождения; группа; наличие золотой или серебряной медали. Вывести данные из файла на экран. Подсчитать количество медалистов. Создать новый файл данных, содержащий сведения о студентах, имеющих золотую медаль (используя существующий файла данных).
10. Создать файл, содержащий следующие данные: ФИО рабочего; количество отработанных дней; сумма зарплаты. Вывести данные из файла на экран. Подсчитать общую сумму зарплаты, выданной рабочим. Создать новый файл данных, содержащий сведения о рабочих, зарплата которых менее заданного значения.

Название задачи

Дается краткое определение решаемой задачи, название программного комплекса, указывается система программирования для его реализации и требования к аппаратному обеспечению (компьютеру, внешним устройствам и т. д.).

Описание

Подробно излагается постановка задачи, описывается применяемая математическая модель для задач вычислительного характера, метод обработки входных данных для задач не вычислительного (логического) характера и т.д.

Управление режимами работы программы

Формулируются основные требования к способу взаимодействия пользователя с программой (интерфейс пользователь–компьютер).

Входные данные

Описываются входные данные, указываются пределы, в которых они могут изменяться, значения, которые они не могут принимать, и т.д.

Выходные данные

Описываются выходные данные, Указывается, в каком виде они должны быть представлены — в числовом или текстовом, в виде таблицы или Web–документа, на диске, печатающем устройстве или дисплее и другие.

Ошибки

Перечисляются возможные ошибки пользователя при работе с программой. Например, ошибки при вводе исходных данных и другие. Указываются способы диагностики (под диагностикой понимается выявление, обнаружение ошибок при работе программного комплекса) и защиты от этих ошибок на этапе проектирования, а также возможная реакция программного комплекса на эти действия.

Контрольные примеры работы программного комплекса

Приводится один или несколько примеров работы программного комплекса, на которых в простейших случаях проводится его отладка и тестирование.

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций:

Преподаватель в процессе подготовки к проведению занятий исходит из того, что

- студенты знакомы с фундаментальными основами и практикой использования средств информационно-коммуникационных технологий, таких как компьютер, периферия, средства связи, системное программное обеспечение, системы программирования, пакеты прикладных программ;
- в ходе проводимых занятий предлагаемые студентам задания, упражнения, и т.п. должны быть ориентированы на расширение познаний обучаемых в области программирования;
- для приобщения обучаемых к поиску, к исследовательской работе, для развития их творческого потенциала следует по возможности избегать прямого руководства работой обучающихся при выполнении ими тех или иных заданий, чаще выступать в роли консультанта, эксперта, коллеги-исследователя;
- оценка ЗУН осуществляется при защите готовой программы.

7.4. Оценочные средств для проведения промежуточной аттестации

Промежуточная аттестация

Форма аттестации	Знания	Умения	Иметь практический опыт	Личные качества студента	Примеры оценочных средств
Экзамен, тест по теме, комплексный тест по предметам	Контроль знания базовых положений в прикладном программировании	Оценка умения понимать специальную терминологию	Оценка навыков логического анализа задачи придумывать алгоритм.	Оценка способности грамотно и четко излагать материал	Вопросы: прилагаются
		Оценка умения разрабатывать алгоритмы и писать программы на языках высокого уровня.	Оценка навыков Переработки алгоритма в компьютерную программу	Оценка способности грамотно и четко излагать ход работы программы и аргументировать результаты	Задачи прилагаются

7.4.1. Примерные вопросы для проведения аттестации

Вопросы к экзамену

1. Понятие и сущность инструментального программного обеспечения;
2. Виды инструментального обеспечения разработки программ;
3. Понятие и типовой состав интегрированной среды разработки;
4. Обзор наиболее популярных инструментальных сред разработки;
5. Визуальное программирование. Технология Drag and Drop.
6. Визуальная среда Delphi;
7. Классификация ошибок;
8. Обработка исключительных ситуаций;
9. Классы исключений;
10. Тестирование и отладка программы;
11. Классы и объекты Object Pascal;
12. Основные свойства ООП в Delphi;
13. Понятие и структура проекта Delphi;
14. Базовые конструкции языка Delphi.

15. Условные операторы и операторы выбора.
16. Операторы циклов.
17. Основные этапы создания компонент;
18. Создание модуля компонента;
19. Классификация ошибок возникающих в программе;
20. Обработка исключительных ситуаций;
21. Классы исключений;
22. Понятие и виды подпрограмм.
23. Особенности описания и использования процедур и функций;
24. Создание и использование библиотек подпрограмм;
25. Понятие DLL – библиотек.
26. Структура проекта DLL.
27. Понятие потоков и процессов.
28. Средства синхронизации потоков;
29. Создание модулей – потоков в Delphi;
30. Компоненты, используемые для работы с БД;
31. Визуальные компоненты отображения данных;
32. Подключение и отображение наборов данных;
33. Создание консольных приложений в Delphi.
34. Динамические переменные;
35. Использование списков;
36. Создание файла справочной информации.
37. Создание проекта справочной системы (HLP - файла);
38. Этапы создания справочной системы (CHM - файла);
39. Основные этапы создания инсталлятора программы.

7.4.2. Примерные экзаменационные задачи

Балльно-рейтинговая система контроля

В основу фиксирования достижений учащихся положена компьютерная программа «**Рейтинг-автомат**», разработанная Левиным Л.Л., в которой имеются две главные связанные таблицы: 1. Список студентов по горизонтали и по вертикали Дата/Код КИМ/Сложность (вес) КИМ. 2. Список кодов тем занятий (КИМ) и содержание. При заполнении таблиц происходит автоматический пересчет баллов, набранных каждым студентом, ранжирование списка по набранным баллам, печать индивидуальных экзаменационных билетов с перечислением тем, пропущенных студентом. За посещение студентом занятия ему начисляется 1 балл. В качестве КИМ в значительной степени используются различные компьютерные тестовые системы, набранные баллы из которых заносятся в первую таблицу. Сложность КИМ назначает преподаватель.

Для текущего и итогового контроля применяется компьютерная программа «**ЭкзамL**», разработанная Левиным Л.Л. для компьютерного контроля и тестирования, работающая в режиме обучения и контроля. Характеристики тем, заложенных в программу, приведены ниже.

1. Логические основы языков программирования. Паскаль 67 вопросов
2. Алгоритмизация задач. 54 вопроса
3. Pascal Ввод данных, оператор присваивания, вывод данных 37 вопросов

4. Pascal Стандартные функции, использование. Основные конструкции 74 вопросов
5. Файлы типизированные, текстовые, нетипизированные 28 вопросов
6. Delphi. Компоненты, свойства, события 81 вопрос
7. Delphi Стандартные функции 30 вопросов
8. Delphi Проект диалоговые окна графика 36 вопросов
9. Delphi Базы данных, Язык SQL 27 вопросов
10. Delphi Техник-программист Квалификационные тесты 114 вопросов
11. Базы данных Технология хранения, поиска и сортировки информации 27 вопросов
12. Технология ПО. Этапы разработки 86 вопросов
13. Технология ПО. Автоматизация и методы программирования 105 вопросов
14. Windows Файловая система 51 вопрос
15. Архитектура компьютера 50 вопросов
16. Информационные технологии в образовании 100 вопросов
17. Операционные системы 83 вопросов

8. ДОПОЛНИТЕЛЬНОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Приложение 1. Краткий конспект лекционных занятий

Инструментальные средства разработки программ

Выделены и охарактеризованы основные этапы разработки программного обеспечения. Для каждого этапа приведены и описаны средства, которые могут быть применены для достижения целей этапа.

1. Терминология

Прежде чем приступить к рассмотрению средств разработки, которые могут быть применены для создания программ, необходимо определиться с основными понятиями, терминами, которые будут использоваться в статье. В соответствии с тематикой статьи базовым термином для нас, конечно же, является «средства разработки программ». Применительно к области разработки программного обеспечения данное определение может звучать следующим образом:

Средства разработки программного обеспечения – *совокупность приемов, методов, методик, а также набор инструментальных программ (компиляторы, прикладные/системные библиотеки и т.д.), используемых разработчиком для создания программного кода Программы, отвечающего заданным требованиям.*

С учетом данного определения термин «Разработка программ» будет звучать следующим образом:

Разработка программ – *сложный процесс, основной целью которого является создание, сопровождение программного кода, обеспечивающего необходимый уровень надежности и качества. Для достижения основной цели разработки программ используются средства разработки программного обеспечения.*

2. Основные средства, используемые на разных этапах разработки программ

В зависимости от предметной области и задач, поставленных перед разработчиками, разработка программ может представлять собой достаточно сложный, поэтапный процесс, в котором задействовано большое количество участников и разнообразных средств. Для того, чтобы определить, когда и в каких случаях какие средства применяются, выделим основные этапы разработки программного обеспечения. Наибольший интерес для проблематики рассматриваемого вопроса представляют следующие этапы разработки:

1. Проектирование приложения.
2. Реализация программного кода приложения.
3. Тестирование приложения.

Здесь сознательно опущены этапы, связанные с написанием технического задания, планирования сроков, бюджета и т.д. Причина этого заключается в том, что на данных этапах, за редким исключением, практически не используются специфические средства разработки.

2.1 Средства проектирования приложений

На этапе проектирования приложения в зависимости от сложности разрабатываемого программного продукта, напрямую зависящего от предъявляемых требований, выполняются следующие задачи проектирования:

1. Анализ требований.
2. Разработка архитектуры будущего программного обеспечения.
3. Разработка устройств основных компонент программного обеспечения.
4. Разработка макетов Пользовательских интерфейсов.

Результатом проектирования обычно является «Эскизный проект» (Software Design Document) или «Технический проект» (Software Architecture Document). Задача «Анализ требований» обычно выполняется с использованием методов системологии (анализа и синтеза) с учетом экспертного опыта проектировщика. Результатом анализа обычно является содержательная или формализованная модель процесса функционирования программы. В зависимости от сложности процесса для построения данных моделей могут быть применены различные методы и вспомогательные средства. В общем случае для описания моделей обычно применяются следующие нотации (в скобках приведены программные средства, которые могут быть использованы для получения моделей):

- BPMN (Vision 2003 + BPMN, AcuaLogic BPMN, Eclipse, Sybase Power Designer).
- Блок-схемы (Vision 2003 и многие другие).
- ER-диаграмы (Visio 2003, ERWin, Sybase Power Designer и многие другие).
- UML-диаграмы (Sybase Power Designer, Rational Rose и многие другие).
- макеты, мат-модели и т.д.

Иногда, когда разрабатываемый программный продукт предназначен для автоматизации какой-либо сложной деятельности задача Анализа (Моделирования) выполняется до составления технических требований к будущему продукту. Результаты анализа позволяют сформировать обоснованные требования к той или иной функциональности разрабатываемой программы и просчитать реальную выгоду от внедрения разрабатываемого продукта. Более того, иного получается так, что по результатам анализа первоначальные цели и задачи автоматизации кардинально меняются или по результатам оценки эффективности разработки и внедрения принимается решение продукт не разрабатывать.

Целью второй и третьей задачи из приведенного списка задач является разработка модели (описания) будущей системы, понятной для кодировщика – человека, который пишет код программы. Здесь огромное значение имеет то, какую парадигму программирования (парадигму программирования также необходимо рассматривать как средство разработки) необходимо использовать при написании программы. В качестве примера основных парадигм необходимо привести следующее:

- Функциональное программирование;
- Структурное программирование;
- Императивное программирование;
- Логическое программирование;
- Объектно-ориентированное программирование (прототипирование; использование классов; субъективно-ориентированное программирование **).**

Выбор её во многом зависит от сложившихся привычек, опыта, традиций, инструментальных средств, которыми располагает коллектив разработчиков. Иногда разрабатываемый программный продукт настолько сложен, что для решения ряда задач в разных компонентах системы используются разные парадигмы. Необходимо отметить, что выбор того или иного подхода накладывает ограничения на средства, которые будут применены на этапе реализации программного кода. Результатом решения данной задачи в зависимости от подхода могут быть (в скобках приведены программные средства, которые могут быть использованы для их получения):

- диаграмма классов и т.д (Ration Rose, Sybase PowerDisigner и многие другие).
- описание модулей структур и их программного интерфейса (например, Sybase PowerDisigner и многие другие).

Разработка макетов пользовательских интерфейсов подразумевает создание наглядного представления того, как будут выглядеть те или иные видеоформы, окна в разрабатываемом приложении. Решение данной задачи основывается на применение средств дизайнера, которые в данной статье рассматриваться не будут.

2.2 Средства реализации программного кода

На этапе реализации программного кода выполняется кодирование отдельных компонент программы в соответствии с разработанным техническим проектом. Средства, которые могут быть применены, в значительной степени зависит от того, какие подходы были использованы во время проектирования и, кроме этого, от степени проработанности технического проекта. Тем не менее, среди средств разработки программного кода необходимо выделить следующие основные виды средств (в скобках приведено примеры средств): • методы и методики алгоритмизации.

- языки программирования (Pascal, Delphi. C++,Си, Java, C#, php и другие);
- средства создания пользовательского интерфейса (MFC, WPF, QT, GTK+ и т.д.)
- средства управления версиями программного кода (cvs, svn, VSS).
- средства получения исполняемого кода (MS Visual Studio, gcc и многие другие).
- средства управления базами данных (Oracle, MS SQL, FireBird, MySQL и многие другие).
- отладчики (MS Visual Studio, gdb и т.д.).

2.3 Средства тестирования программ

Основными задачами тестирования является проверка соответствия функциональности разработанной программы первоначальным требованиям, а также выявление ошибок, которые в явном или неявном виде проявляются во время работы программы. Среди основных работ по тестированию можно выделить следующее:

- Тестирование на отказ и восстановление.
- Функциональное тестирование.
- Тестирование безопасности.
- Тестирование взаимодействия.
- Тестирование процесса установки.
- Тестирование удобства пользования.
- Конфигурационное тестирование.
- Нагрузочное тестирование.

Среди основных видов средств, которые могут быть применены для выполнения поставленных работ можно привести следующие:

- средства анализа кода, профилирования (Code Wizard – ParaSoft, Purify – Rational Softawre. Test Coverage – Semantic и т.д.);
- средства для тестирования функциональности (TEST – Parasoft, QACenter – Compuware, Borland SilkTest и т.д.);
- средства для тестирования производительности (QACenter Performance – Compuware и т.д).

3. Заключение

Процесс разработки программ является сложным процессом и то, какие средства необходимо применять во многом зависит от задач, поставленным перед разработчиками. В независимости от задач разработки средства нельзя ограничивать лишь набором каких-то инструментальных средств, также необходимо включать

методы, методики, подходы и все-то, что применяется для создания программы, отвечающей заданным требованиям.

Словарь. Инструментальные среды

макрокоманда Строка кода исходной программы, которая инициирует процедуру формирования строки кода объектной программы

преобразование программ трансформация программ Преобразование спецификации алгоритма (программы) с целью ее распараллеливания, повышения ее эффективности, устранение уязвимостей и др. при сохранении неизменной семантики спецификации

макрос Оператор в программе, заменяемый на некоторый текст, получаемый с учетом его фактических параметров

workbench Программная платформа для разработки и интеграции программных продуктов, прикладных пользовательских программ в едином информационном пространстве проекта

эмулятор
Устройство или программа, имитирующие работу другого устройства или программы

метод "водопада" водопадный метод Метод разработки программного обеспечения, заключающийся в последовательном выполнении этапов определения требований, проектирования, кодирования, тестирования и развертывания программной системы

итеративная разработка
Метод разработки программного обеспечения, при котором на каждом последовательном этапе разработки выполняются процедуры определения требований, проектирования, кодирования и тестирования соответствующей части программного продукта. Следовательно, происходит итерационное продвижение к окончательному результату, причем каждое промежуточное решение уже является продуктом, способным выполнять определенные запланированные функции

архитектура, управляемая моделью Model Driven Architecture MDA
Предлагаемая организацией OMG технология разработки программного обеспечения, основанная на выполнении всех проектных процедур на уровне моделей с последующей автоматической генерацией кода. Разработка, управляемая моделями заключается в том, что все основные проектные решения принимаются при оперировании платформо-независимыми моделями (PIM), а реализация модели в виде продукта становится формальной процедурой преобразования PIM в платформо-специфическую модель (PSM) и далее в программный код

разработка, управляемая моделями разработка программ, управляемая моделями Model Driven Design MDD
Процесс проектирования программ, сводящийся к ручной разработке моделей и автоматической генерации кода по имеющимся моделям.

управление конфигурациями и изменениями конфигурационное управление Software Configuration Management Методология управления разработкой программного обеспечения, основанная на синхронизации действий разработчиков, контроле внесения изменений в проект, управлении единым репозитарием проекта

плагин Независимо компилируемый программный модуль, динамически подключаемый к основной программе, предназначенный для расширения и/или использования её возможностей

виджет Примитив графического интерфейса пользователя, имеющий стандартный внешний вид

Iintegrated Development Environment IDE интегрированная инструментальная среда интегрированная среда разработки Интегрированная среда разработки приложений, в частности, программного обеспечения

SDK Software Development Kit Набор средств разработки программного обеспечения, утилит и документации, который позволяет программистам создавать приложения по определённой технологии или для определённой платформы (программной или программно-аппаратной)

MFC Microsoft Foundation Classes Базовый набор (библиотека) классов, написанных на языке C++ и предназначенных для упрощения и ускорения процесса программирования для Windows

Delphi Среда программирования на основе операционной системы Windows, визуального пользовательского интерфейса, объектно-ориентированного языка Object Pascal (называемого также Delphi) и средств доступа к базам данных

VisualAge Обьектно-ориентированная среда программирования на базе языка Smalltalk для разработки приложений для архитектуры клиент-сервер

Software Development Platform SDP
Среда разработки программного обеспечения, созданная корпорацией IBM

Eclipse Инвариантная интегрированная среда разработки программного обеспечения (IDE), состоящая из ядра и плагинов. С ее помощью могут создаваться веб-сайты, встраиваемые Java-программы, программы на C++ и компоненты JavaBeans

Eclipse Modeling Framework EMF Среда разработки программ на основе управления моделями (Model-Driven Development) в системе Eclipse.

Code::Blocks Свободно распространяемая кроссплатформенная среда разработки программного обеспечения. Написана на Си++, может масштабироваться за счёт подключаемых модулей. Поддерживает языки программирования C, C++, D (с ограничениями).

PowerBuilder Объектно-ориентированная инструментальная среда для разработки графических клиент/серверных приложений, имеющих доступ к базам данных и другой корпоративной информации, хранящейся локально или на сетевых серверах

OSGi Open Services Gateway Initiative Инструментальная среда создания Java-приложений на основе динамической многомодульной (плагинной) шины, разрабатываемая консорциумом OSGi Alliance.

команда разработчиков В команде разработчиков сложных программных проектов имеются члены, выполняющие роли менеджера проекта, аналитиков, архитекторов, разработчиков, тестировщиков, менеджеров по развертыванию

Standard Widget Toolkit SWT Библиотека программных средств для разработки графических интерфейсов пользователя на языке Java

Среда Microsoft.NET Microsoft.NET .NET Среда разработки и выполнения программного обеспечения, созданная компанией Microsoft.NET

Framework NET.Framework Часть Microsoft.NET, служащая для создания приложений

Common Language Runtime CLR Часть .NET Framework, обеспечивающая совместное выполнение приложений .NET, в том числе разноязыковых

Common Language Specification CLS Набор правил, которых должен придерживаться компилятор языка при создании .NET-приложений, запускаемых в среде CLR

библиотека классов .NET Framework Библиотека классов в среде .NET Framework Microsoft Intermediate Language MSIL Промежуточный язык, служащий для облегчения перевода разноязыковых приложений в среду .NET JIT JITter

Компилятор с языка MSIL на машинный язык

PE файл Portable Executable File Программный компонент, который может непосредственно исполняться или использоваться повторно

сборка assembly Программный компонент (пакет), состоящий из одного или нескольких PE файлов или компонентов DLL
развертывание Процедура объединения компонентов и сборок в совместно исполняемую распределенную систему программ

Приложение 2. Презентации.

Папка F:\Uni\2017_18\14П\Инструментальные средства разработки ПО\РП\2017\Лекции Презентации На 30.08.2017

№	Имя файла	Байт	Дата
1	Информатика Программное обеспечение.ppt	625664	09.08.2011
2	Лекции Инструментальные средства.doc	653824	05.09.2017
3	Программное обеспечение.ppt	1785856	05.09.2017
4	Системное_ПО.ppt	313856	05.09.2017
5	Средства разработки информационных систем.pptx	468201	10.04.2013

ЛИСТ

изменений рабочей учебной программы по дисциплине

МДК.03.02. Инструментальные средства разработки программного обеспечения

Дополнения и изменения, вносимые в рабочую программу дисциплины

Основания внесения дополнений и изменений	Раздел РПД, в который вносятся изменения	Содержание вносимых дополнений, изменений
Предложение работодателя		
Предложение составителя программы		
Другие основания		

Составитель: преподаватель _____ Л.Л. Левин канд.техн.наук
подпись

Утвержден на заседании предметной (цикловой) комиссии профессиональных дисциплин специальности Программирование в компьютерных системах
протокол № __1__ от «_30_» _августа__ 2019 г.

Председатель предметной (цикловой) комиссии профессиональных дисциплин специальности Программирование в компьютерных системах

_____ Л.А. Благова « 30» августа 2019 г.

Зам. директора по УР филиала _____ Т. А. Резуненко
« 30 » августа 2019 г.

Заведующая сектором библиотеки _____ Л. Г. Соколова
« 30 » августа 2019 г.

Инженер-электроник (программно-информационное обеспечение образовательной программы) _____ А. В. Сметанин
« 30 » августа 2019 г.

РЕЦЕНЗИЯ

на рабочую программу по учебной дисциплине **МДК.03.02. «Инструментальные средства программного обеспечения»** по специальности **09.02.03 «Программирование в компьютерных системах» СПО,** разработанную кандидатом техн. наук, преподавателем Левиным Львом Львовичем.

Структура программы соответствует современному уровню развития технологий программирования и компьютерной техники, она включает в себя описание актуального программного обеспечения, алгоритмических и технических средств.

Рабочая программа предусматривает освоение профессиональных компетенций: (ПК 3.1.- 3.6.) и видов деятельности, в соответствии с ФГОС № 804 от «28» июля 2014 г.

В программу включено изучение средств визуального проектирования и разработки прикладных, с проработкой интерфейса, необходимого инструментария. Программа имеет достаточную степень полноты и законченности изучения предмета в условиях СПО.

Дисциплина «Инструментальные средства программного обеспечения» предусматривает приобретение навыков в создании программных продуктов с применением средств интеграции сторонних разработчиков.

В программе нашли отражение важные примеры использования инструментов для создания инсталляторов программ и систем помощи пользователю, что даёт возможность и умение решать конкретные прикладные задачи, использовать имеющиеся инструменты.

Структура программы соответствует современным требованиям. Содержание каждого её элемента разработано с достаточной степенью полноты и законченности. Пояснительная записка раскрывает ведущие цели программы, включает в себя краткую характеристику её предметного содержания.

Программа содержит необходимый список учебных пособий.

Применение в учебном процессе балльно- рейтинговой системы оценивания и программ тестирования заслуживает положительной оценки.

В целом анализируемая программа учебной дисциплины заслуживает положительной оценки, поскольку она достаточно продумана и ориентирована на подготовку обучающихся к использованию полученных навыков в своей профессиональной деятельности.

Таким образом, рабочая программа содержит все необходимые элементы рекомендуемой структуры, обладает достаточной полнотой и законченностью, является полезным практическим документом при преподавании дисциплины «Инструментальные средства программного обеспечения» и в обучении студентов.

Рецензент: Сметанин Алексей Владиславович, ООО "Современные Информационные Технологии" г. Геленджик, директор



РЕЦЕНЗИЯ

на рабочую программу по учебной дисциплине

МДК.03.02. «Инструментальные средства программного обеспечения»
по специальности 09.02.03 «Программирование в компьютерных системах»

СПО, разработанную канд.техн. наук, преподавателем **Левиным Львом Львовичем.**

По структуре программа соответствует современному уровню развития технологий программирования и компьютерной техники, она включает в себя описание актуального программного обеспечения, алгоритмических и технических средств, а также средств оценивания успеваемости.

Рецензируемая рабочая программа предусматривает освоение профессиональных компетенций: (ПК 3.1.- 3.6.) и видов деятельности, согласно ФГОС № 804 от «28» июля 2014 г.

Программа содержит изучение средств объектно-ориентированного и визуального проектирования и разработки прикладных программ, с проработкой интерфейса, необходимого инструментария. Программа имеет достаточную степень полноты и законченности в условиях СПО.

Дисциплина «Инструментальные средства программного обеспечения» предусматривает приобретение навыков в создании программных продуктов с применением различных инструментальных средств.

В рабочей программе отражены важные примеры применения инструментов для разработки программ, создания инсталляторов программ и систем помощи пользователю, что позволяет решать актуальные прикладные задачи, использовать имеющиеся инструменты.

По структуре программа соответствует современным требованиям. Содержание её элементов проработано с достаточной степенью полноты и законченности. Текст программы раскрывает значимые цели программы содержат краткую характеристику её предметного содержания.

В программе приводится необходимый список учебных пособий.

Следует отметить применение в учебном процессе балльно- рейтинговой системы оценивания знаний и программ тестирования.

По существу рецензируемая программа учебной дисциплины заслуживает положительной оценки, она достаточно продумана и ориентирована на подготовку обучающихся к использованию полученных навыков в своей профессиональной деятельности.

В заключение отметим, что рабочая программа содержит все необходимые элементы рекомендуемой структуры, обладает достаточной полнотой и законченностью, является полезным практическим документом в обучении студентов и преподавании дисциплины «Инструментальные средства программного обеспечения».

Рецензент: Кривописенко Татьяна Петровна, системный администратор ЗАО «Геленджикский дельфинарий»

