



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение
высшего образования

«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
ФГБОУ ВО «ИГУ»

Кафедра радиофизики и радиоэлектроники



Декан

Буднев Н.М.

«22» апреля 2020 г.

Рабочая программа дисциплины (модуля)

Наименование дисциплины (модуля) **Б1.В.ДВ.01.01 Объектно-ориентированное программирование**

Направление подготовки 10.03.01 Информационная безопасность

Тип образовательной программы бакалавриат

Направленность (профиль) подготовки №4 Безопасность автоматизированных систем (в сфере профессиональной деятельности)

Квалификация выпускника бакалавр

Форма обучения очная

Согласовано с УМК физического факультета

Протокол № 25 от «21» апреля 2020 г.
Председатель _____ Буднев Н.М.

Рекомендовано кафедрой радиофизики и радиоэлектроники:

Протокол № 8
От «20» марта 2020 г.
И.О.Зав. кафедрой _____ Колесник С.Н.

Иркутск 2020 г.

Содержание

	стр.
1. Цели и задачи дисциплины (модуля)	3
2. Место дисциплины в структуре ОПОП	3
3. Требования к результатам освоения дисциплины (модуля)	3
4. Объем дисциплины (модуля) и виды учебной работы	5
5. Содержание дисциплины (модуля)	5
5.1. Содержание разделов и тем дисциплины (модуля)	5
5.2. Разделы дисциплины и междисциплинарные связи с обеспечиваемыми (последующими) дисциплинами.....	7
5.3. Разделы и темы дисциплин (модулей) и виды занятий	7
6. Перечень семинарских, практических занятий и лабораторных работ	7
6.1. План самостоятельной работы студентов.....	8
6.2. Методические указания по организации самостоятельной работы студентов	9
7. Примерная тематика курсовых работ (проектов)	9
8. Учебно-методическое и информационное обеспечение дисциплины (модуля):	10
а) основная литература	10
б) базы данных, информационно-справочные и поисковые системы.....	10
9. Материально-техническое обеспечение дисциплины (модуля)	10
10. Образовательные технологии	10
11. Оценочные средства (ОС):	11
11.1. Оценочные средства для входного контроля	11
11.2. Оценочные средства текущего контроля.....	11
11.3. Оценочные средства для промежуточной аттестации.....	15

1. Цели и задачи дисциплины (модуля)

Дисциплина «Объектно-ориентированное программирование», как дисциплина профессионального цикла направлена на достижение следующих целей:

- подготовку специалистов к деятельности, связанной с разработкой, сопровождением и модернизацией программных систем, с использованием современных объектно-ориентированных технологий и средств разработки программного обеспечения

Для достижения поставленной цели сформулированы следующие задачи:

- овладение основными понятиями объектно-ориентированного программирования;
- изучение принципов программирования приложений, ориентированных на применение иерархии классов и библиотек;
- освоение принципов разработки и моделирования объектов, с использованием объектно-ориентированного языка программирования;
- Овладение инструментами интегрированной среды разработки программ на объектно-ориентированном языке программирования.

2. Место дисциплины в структуре ОПОП

Дисциплине «Объектно-ориентированное программирование» отводится существенная роль в профессиональной подготовке будущего специалиста. В процессе изучения курса студенты должны получить знания по основополагающим принципам разработки программных систем. В процессе обучения студенты получают знания по основным средствам моделирования, программирования, а также по использованию различных методик разработки современных сложных программных систем.

Целью дисциплины является приобретение базовых знаний и навыков проектирования и разработки приложений с применением объектно-ориентированного подхода.

В состав задач изучения дисциплины входят:

1. Изучение теоретических основ объектного представления информации.
2. Изучение способов представления информации в конкретных прикладных системах.

3. Требования к результатам освоения дисциплины (модуля)

Процесс изучения дисциплины (модуля) направлен на формирование следующих компетенций:

- способность принимать участие в организации и сопровождении аттестации объекта информатизации по требованиям безопасности информации (ПК-5);
- способность проводить анализ исходных данных для проектирования подсистем и средств обеспечения информационной безопасности и участвовать в

проведении технико-экономического обоснования соответствующих проектных решений (ПК-7);

- способностью оформлять рабочую техническую документацию с учетом действующих нормативных и методических документов (ПК-8)

В результате изучения дисциплины студент должен:

Знать:

- историю развития, состояние и тенденции развития языков программирования;
- основы обработки событий,
- принципы построения объектно-ориентированных алгоритмов,
- основы управления объектами,
- основные инструменты разработки интерфейса программных средств,

Уметь:

- формализовать поставленную задачу;
- применять полученные знания в различных предметных областях;
- создавать классы, объявлять методы и свойства моделируемых объектов;
- разрабатывать интерфейс программы, с использованием библиотек классов и элементов управления;
- проектировать иерархию классов в объектно-ориентированной программе;
- обрабатывать события, с использованием виртуальных методов обработки событий;
- производить отладку программ, основанных на объектно-ориентированном подходе.

Владеть:

- навыками работы с компьютером, а также с современными интегрированными средами разработки программных систем и приложений.

4. Объем дисциплины (модуля) и виды учебной работы

Вид учебной работы	Всего часов / зачетных единиц	Семестры			
		4	-	-	-
Аудиторные занятия (всего)	64/1,78	64/1,78	-	-	-
В том числе:	-	-	-	-	-
Лекции	20/0,56	20/0,56	-	-	-
Практические занятия (ПЗ)	-	-	-	-	-
Семинары (С)	-	-	-	-	-
Лабораторные работы (ЛР)	40/1,11	40/1,11	-	-	-
Контроль самостоятельной работы (КСР)	4/0,11	4/0,11			
Самостоятельная работа (всего)	80/2,22	80/2,22	-	-	-
В том числе:	-	-	-	-	-
Курсовой проект (работа)					
Расчетно-графические работы					
Реферат (при наличии)					
<i>Другие виды самостоятельной работы</i>	80/2,22	80/2,22			
Вид промежуточной аттестации (<i>зачет, экзамен</i>)	зачет	зачет			
Контактная работа (всего)	72/2	72/2			
Общая трудоемкость	часы	144	144		
	зачетные единицы	4	4		

5. Содержание дисциплины (модуля)

5.1. Содержание разделов и тем дисциплины (модуля)

РАЗДЕЛ 1. (Тема 1). История ООП. Объектно-ориентированные и процедурные языки. Идеология и методические основы ООП. Основные понятия ООП.

История и эволюция языков программирования. Классификация языков программирования. Процедурные языки. Объектно-ориентированные языки. Основные постулаты ООП. Классы, объекты. Свойства и методы объектов.

РАЗДЕЛ 2. (Тема 2). Операционная система Windows. Средства разработки приложений под ОС Windows. Графический интерфейс пользователя.

Обзор ОС Windows. Основные понятия ОС, окна, приложения, процессы. Средства разработки приложений для Windows на различных языках программирования. Язык C++.

Средства проектирования и моделирования графического интерфейса пользователя.

РАЗДЕЛ 3. (Тема 3). Визуально-событийное программирование в ОС Windows.**Классификация сообщений**

Основы обработки событий. Реализация обработчиков событий в приложениях. Системные и пользовательские события. Механизм передачи событий между компонентами программы и обработчиками. Основные элементы управления Windows и генерируемые ими события.

РАЗДЕЛ 4. (Тема 4). Структура приложений в ОС Windows. Библиотека классов.**Типы создаваемых приложений**

Структура проектов на C++. Основные компоненты проектов. Типы приложений Windows

и особенности их разработки. Консольные приложения. Оконные приложения. Библиотеки. Управление проектом. Виды и способы управления ресурсами приложений.

Библиотека классов MFC. Архитектура «Документ - Представление». Многооконные и однооконные приложения. Основы сериализации и управления данными документа. Управление видами.

РАЗДЕЛ 5. (Тема 5). Проектирование и разработка интерфейса приложения.

Основные инструменты разработки интерфейса. Элементы управления. Способы создания элементов управления. Программное взаимодействие элементов управления в приложении. Передача данных и управляющих команд между элементами интерфейса.

Основные классы элементов управления. Создание собственных классов элементов интерфейса приложения.

РАЗДЕЛ 6. (Тема 6). Графика и анимация в приложениях

Основные понятия графики. Физическая и логическая системы координат. Основные классы контекстов устройств рисования в Windows. Средства рисования библиотеки MFC. Функции рисования и отображения объектов. Анимация. Работа с различными форматами рисунков.

5.2. Разделы дисциплины и междисциплинарные связи с обеспечиваемыми (последующими) дисциплинами

№ п/п	Наименование обеспечиваемых (последующих) дисциплин	№№ разделов (тем) данной дисциплины, необходимых для изучения обеспечиваемых (последующих) дисциплин
1	Безопасность компьютерных сетей	1-6
2	Практика по получению первичных профессиональных умений и навыков	1-6
3	Проектно-технологическая практика	1-6

5.3. Разделы и темы дисциплин (модулей) и виды занятий

№ п/п	Наименование раздела	Наименование темы	Виды занятий в часах					Всего
			Лекц.	Практ. зан.	Семина	Лаб. зан.	СРС	
1.	Раздел 1	Тема 1	2			4	10	16
2.	Раздел 2	Тема 2	4			8	10	22
3.	Раздел 3	Тема 3	2			6	10	18
4.	Раздел 4	Тема 4	4			8	20	32
5.	Раздел 5	Тема 5	4			8	20	32
6.	Раздел 6	Тема 6	4			6	10	20

6. Перечень семинарских, практических занятий и лабораторных работ

№ п/п	№ раздела и темы дисциплины (модуля)	Наименование семинаров, практических и лабораторных работ	Трудоемкость (час.)	Оценочные средства	Формируемые компетенции
1	2	3	4	5	6
1.	Раздел 1.	Лабораторная работа №1	2	Тестовый контроль по теме	ПК-5
2.	Раздел 1.	Лабораторная работа №2	2	Тестовый контроль по теме	ПК-5
3.	Раздел 2.	Лабораторная работа №3	2	Тестовый контроль по теме	ПК-5
4.	Раздел 2.	Лабораторная работа №4	2	Тестовый контроль по теме	ПК-7
5.	Раздел 2.	Лабораторная работа №5	2	Тестовый контроль по теме	ПК-7
6.	Раздел 2.	Лабораторная работа №6	2	Тестовый контроль по теме	ПК-8

7.	Раздел 3.	Лабораторная работа №7	3	Тестовый контроль по теме	ПК-5
8.	Раздел 3.	Лабораторная работа №8	3	Тестовый контроль по теме	ПК-5
9.	Раздел 4.	Лабораторная работа №9	2	Тестовый контроль по теме	ПК-7
10.	Раздел 4.	Лабораторная работа №10	2	Тестовый контроль по теме	ПК-8
11.	Раздел 4.	Лабораторная работа №11	2	Тестовый контроль по теме	ПК-5
12.	Раздел 4.	Лабораторная работа №12	2	Тестовый контроль по теме	ПК-7
13.	Раздел 5.	Лабораторная работа №13	2	Тестовый контроль по теме	ПК-7
14.	Раздел 5.	Лабораторная работа №14	2	Тестовый контроль по теме	ПК-8
15.	Раздел 5.	Лабораторная работа №15	2	Тестовый контроль по теме	ПК-8
16.	Раздел 5.	Лабораторная работа №16	2	Тестовый контроль по теме	ПК-8
17.	Раздел 6.	Лабораторная работа №17	3	Тестовый контроль по теме	ПК-7
18.	Раздел 6.	Лабораторная работа №18	3	Тестовый контроль по теме	ПК-8

6.1. План самостоятельной работы студентов

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
1-7	1-3	Подготовка к контрольной работе №1	№1	Учебный сайт	30
8		Контрольная работа №1.		Учебный сайт	
9		Подведение итогов по контрольной работе №1. Работа над ошибками по контрольной работе №1.		Учебный сайт	
10-16	4-6	Подготовка итоговой зачетной работы	№2	Учебный сайт	50
17		Подготовка доклада с презентацией		Учебный сайт	
18		Подведение итогов		Учебный сайт	

6.2. Методические указания по организации самостоятельной работы студентов

Текущая самостоятельная работа по дисциплине «Объектно-ориентированное программирование», направленная на углубление и закрепление знаний студента, на развитие практических умений, включает в себя следующие виды работ:

- работа с лекционным материалом;
- подготовка к практическим занятиям;
- выполнение индивидуальных проектов;
- подготовка к контрольным работам;
- подготовка к зачету.

Творческая проблемно-ориентированная самостоятельная работа по дисциплине «Объектно-ориентированное программирование», направленная на развитие интеллектуальных умений, общекультурных и профессиональных компетенций, развитие творческого мышления у студентов, включает в себя следующие виды работ по основным проблемам курса:

- поиск, анализ, структурирование информации;
- выполнение работ по программированию моделей и алгоритмов, обработка и анализ данных;
- участие в конференциях, олимпиадах и конкурсах.

Оценка результатов самостоятельной работы организуется как единство двух форм: самоконтроль и контроль со стороны преподавателя.

Самоконтроль зависит от определенных качеств личности, ответственности за результаты своего обучения, заинтересованности в положительной оценке своего труда, материальных и моральных стимулов, от того насколько обучаемый мотивирован в достижении наилучших результатов. Задача преподавателя состоит в том, чтобы создать условия для выполнения самостоятельной работы (учебно-методическое обеспечение), правильно использовать различные стимулы для реализации этой работы (рейтинговая система), повышать её значимость, и грамотно осуществлять контроль самостоятельной деятельности студента (фонд оценочных средств).

7. Примерная тематика курсовых работ (проектов)

Курсовые работы (проекты) учебным планом не предусмотрены.

8. Учебно-методическое и информационное обеспечение дисциплины (модуля):

а) основная литература

1. Объектно-ориентированное программирование [Электронный ресурс] : учебник / И. А. Барков. - Электрон. текстовые дан. - Санкт-Петербург : Лань, 2022. - 700 с. - ЭБС "Лань". - Неогранич. доступ. - ISBN 978-5-8114-3586-9. [URL]: <https://e.lanbook.com/book/206699> (дата обращения: 11.11.2022).

б) базы данных, информационно-справочные и поисковые системы

1. Учебный сайт Лаборатории ТЗИ Физического факультета ИГУ - – Режим доступа: <https://sites.google.com/view/ltzi/>, свободный.

9. Материально-техническое обеспечение дисциплины (модуля)

Компьютерная лаборатория 323а (14 серверов) и лекционная аудитория 225, оснащенные мультимедийными средствами, электронной базой знаний, системой тестирования, выходом в глобальную сеть Интернет. Технические характеристики серверов обеспечивают возможность моделирования необходимого аппаратного обеспечения для работы с современными компьютерными системами хранения и обработки информации.

10. Образовательные технологии

Для достижения планируемых результатов обучения, в дисциплине «Объектно-ориентированное программирование» используются различные образовательные технологии:

Информационно-развивающие технологии, направленные на формирование системы знаний, запоминание и свободное оперирование ими.

Используется лекционно-семинарский метод, самостоятельное изучение литературы, применение новых информационных технологий для самостоятельного пополнения знаний, включая использование технических и электронных средств информации.

Деятельностные практико-ориентированные технологии, направленные на формирование системы профессиональных практических умений при проведении экспериментальных исследований, обеспечивающих возможность качественно выполнять профессиональную деятельность.

Используется анализ, сравнение методов проведения исследований, выбор метода, в зависимости от объекта исследования в конкретной производственной ситуации и его практическая реализация.

Развивающие проблемно-ориентированные технологии, направленные на формирование и развитие проблемного мышления, мыслительной активности, способности видеть и формулировать проблемы, выбирать способы и средства для их решения. Используются виды проблемного обучения: освещение основных проблем объектно-ориентированного подхода при разработке программного обеспечения на лекциях, учебные дискуссии, коллективная деятельность в группах при выполнении лабораторных работ, решение задач повышенной сложности. При этом используются первые три уровня (из четырех) сложности и самостоятельности: проблемное изложение учебного материала преподавателем; создание преподавателем проблемных ситуаций, а обучаемые вместе с ним включаются в их разрешение; преподаватель создает проблемную ситуацию, а разрешают ее студенты в ходе самостоятельной деятельности.

Личностно-ориентированные технологии обучения, обеспечивающие в ходе учебного процесса учет различных способностей обучаемых, создание необходимых условий для развития их индивидуальных способностей, развитие активности личности в учебном процессе. Личностно-ориентированные технологии обучения реализуются в результате индивидуального общения преподавателя и студента при защите лабораторных работ, при выполнении домашних индивидуальных заданий, решении задач повышенной сложности, на плановых и внеплановых консультациях.

11. Оценочные средства (ОС):

11.1. Оценочные средства для входного контроля

Входной контроль (6 вариантов, 5-й семестр), представляет собой перечень из 10 вопросов и заданий. Входной контроль проводится в письменном виде на первом практическом занятии в течение 15 минут. Проверяется уровень входных знаний.

11.2. Оценочные средства текущего контроля

Вопросы к лабораторным занятиям (6 тем, 18 лабораторных работ).

Представляют собой вопросы для отчетов, выполняемых студентами по результатам лабораторной работы. Отчеты по результатам выполнения лабораторных работ позволяют проверить знание теоретического материала, а также навыки выполнения практических и творческих задач, связанных с разработкой программного обеспечения.

Тестовые вопросы по дисциплине Б1.В.ДВ.01.01 «Объектно-ориентированное программирование и моделирование»

Вариант 1.

1. Класс - это:

- любой тип данных, определяемый пользователем
- тип данных, определяемый пользователем и сочетающий в себе данные и функции их обработки
- структура, для которой в программе имеются функции работы с нею

2. Членами класса могут быть

- как переменные, так и функции, могут быть объявлены как private и как public
- только переменные, объявленные как private
- только функции, объявленные как private
- только переменные и функции, объявленные как private
- только переменные и функции, объявленные как public

3. Что называется конструктором?

- метод, имя которого совпадает с именем класса и который вызывается автоматически при создании объекта класса
- метод, имя которого совпадает с именем класса и который вызывается автоматически при объявлении класса (до создания объекта класса)
- метод, имя которого необязательно совпадает с именем класса и который вызывается при создании объекта класса
- метод, имя которого совпадает с именем класса и который необходимо явно вызывать из головной программы при объявлении объекта класса

4. Объект - это

- переменная, содержащая указатель на класс
- экземпляр класса
- класс, который содержит в себе данные и методы их обработки

5. Отметьте правильные утверждения

- конструкторы класса не наследуются
- конструкторов класса может быть несколько, их синтаксис определяется программистом
- конструкторов класса может быть несколько, но их синтаксис должен подчиняться правилам перегрузки функций
- конструктор возвращает указатель на объект
- конструктор не возвращает значение

6. Что называется деструктором?

- метод, который уничтожает объект
- метод, который удаляет объект
- метод, который освобождает память, занимаемую объектом
- системная функция, которая освобождает память, занимаемую объектом

7. Выберите правильные утверждения

- у конструктора могут быть параметры
- конструктор наследуется, но должен быть перегружен
- конструктор должен явно вызываться всегда перед объявлением объекта
- конструктор вызывается автоматически при объявлении объекта
- объявление каждого класса должно содержать свой конструктор
- если конструктор не создан, компилятор создаст его автоматически

8. Выберите правильные утверждения

- деструктор - это метод класса, применяемый для удаления объекта

- деструктор - это метод класса, применяемый для освобождения памяти, занимаемой объектом
- деструктор - это отдельная функция головной программы, применяемая для освобождения памяти, занимаемой объектом
- деструктор не наследуется
- деструктор наследуется, но должен быть перегружен

9. Что называется наследованием?

- это механизм, посредством которого производный класс получает элементы родительского и может дополнять либо изменять их свойства и методы
- это механизм переопределения методов базового класса
- это механизм, посредством которого производный класс получает все поля базового класса
- это механизм, посредством которого производный класс получает элементы родительского, может их дополнить, но не может переопределить

10. Выберите правильное объявление производного класса

- class MoreDetails:: Details;
- class MoreDetails: public class Details;
- class MoreDetails: public Details;
- class MoreDetails: class(Details);

11. Выберите правильные утверждения:

- если элементы класса объявлены как private, то они доступны только наследникам класса, но не внешним функциям
- если элементы класса объявлены как private, то они недоступны ни наследникам класса, ни внешним функциям
- если элементы объявлены как public, то они доступны наследникам класса, но не внешним функциям
- если элементы объявлены как public, то они доступны и наследникам класса, и внешним функциям

12. Возможность и способ обращения производного класса к элементам базового определяется

- ключами доступа: private, public, protected в теле производного класса
- только ключом доступа protected в заголовке объявления производного класса
- ключами доступа: private, public, protected в заголовке объявления производного класса
- ключами доступа: private, public, protected в теле базового класса

Вариант 2.

1. Выберите правильные соответствия между спецификатором базового класса, ключом доступа в объявлении производного класса и правами доступа производного класса к элементам базового

- ключ доступа - public; в базовом классе: private; права доступа в производном классе - protected
- ключ доступа - любой; в базовом классе: private; права доступа в производном классе - нет прав
- ключ доступа - protected или public ; в базовом классе: protected; права доступа в производном классе - protected
- ключ доступа - private; в базовом классе: public; права доступа в производном классе - public

- ключ доступа – любой; в базовом классе: public; права доступа в производном классе – такие же, как ключ доступа

2. Дружественная функция - это

- функция другого класса, среди аргументов которой есть элементы данного класса
- функция, объявленная в классе с атрибутом friend, но не являющаяся членом класса;

- функция, являющаяся членом класса и объявленная с атрибутом friend;

- функция, которая в другом классе объявлена как дружественная данному

3. Выберите правильные утверждения:

- одна функция может быть дружественной нескольким классам
- дружественная функция не может быть обычной функцией, а только методом другого класса

- дружественная функция объявляется внутри класса, к элементам которого ей нужен доступ

- дружественная функция не может быть методом другого класса

4. Шаблон функции - это...

- определение функции, в которой типу обрабатываемых данных присвоено условное обозначение

- прототип функции, в котором вместо имен параметров указан условный тип

- определение функции, в котором указаны возможные варианты типов обрабатываемых параметров

- определение функции, в котором в прототипе указан условный тип, а в определении указаны варианты типов обрабатываемых параметров

5. Выберите правильные утверждения:

- по умолчанию члены класса имеют атрибут private

- по умолчанию члены класса имеют атрибут public;

- члены класса имеют доступ только к элементам public;

- элементы класса с атрибутом private доступны только членам класса

6. Переопределение операций имеет вид:

- имя_класса, ключевое слово operation, символ операции

- имя_класса, ключевое слово operator, символ операции, в круглых скобках могут быть указаны аргументы

- имя_класса, ключевое слово operator, список аргументов

- имя_класса, два двоеточия, ключевое слово operator, символ операции

7. Для доступа к элементам объекта используются:

- при обращении через имя объекта – точка, при обращении через указатель – операция «->»

- при обращении через имя объекта – два двоеточия, при обращении через указатель – операция «точка»

- при обращении через имя объекта – точка, при обращении через указатель – два двоеточия

- при обращении через имя объекта – два двоеточия, при обращении через указатель – операция «->»

8. Полиморфизм – это :

- средство, позволяющее использовать одно имя для обозначения действий, общих для родственных классов

- средство, позволяющее в одном классе использовать методы с одинаковыми именами;

- средство, позволяющее в одном классе использовать методы с разными именами для выполнения одинаковых действий

- средство, позволяющее перегружать функции для работы с разными типами или разным количеством аргументов.

9. Полиморфизм реализован через механизмы:

- перегрузки функций, виртуальных функций, шаблонов
- перегрузки функций, наследования методов, шаблонов;
- наследования методов, виртуальных функций, шаблонов
- перегрузки функций, наследования, виртуальных функций.

10. Виртуальными называются функции:

- функции базового класса, которые могут быть переопределены в производном классе
- функции базового класса, которые не используются в производном классе;
- функции базового класса, которые не могут быть переопределены в базовом классе;
- функции производного класса, переопределенные относительно базового класса

11. Выберите правильный вариант выделения динамической памяти под переменную X типа float:

- float *ptr = new float; X = *ptr;
- float & ptr = new float; X = & ptr;
- float * ptr = &X; X = new float;

12. Полиморфизм в объектно-ориентированном программировании реализуется:

- через механизмы перегрузки (функций и операций), виртуальные функции и шаблоны
- через механизмы перегрузки (функций и операций) и шаблоны;
- через виртуальные функции и шаблоны;
- через механизмы перегрузки (функций и операций) и виртуальные функции

11.3. Оценочные средства для промежуточной аттестации

(в форме зачета).

Проектные задания на разработку программ (4 варианта).

Проверяется степень усвоения теоретических и практических знаний, приобретенных умений на репродуктивном и продуктивном уровне.

Разработчики:



преподаватель

Усенко О.В.

Программа рассмотрена на заседании кафедры радиопизики и радиоэлектроники «20» марта 2020 г.

Протокол № 8 И.О.Зав. кафедрой



Колесник С.Н.

Настоящая программа, не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.