

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 33

УТВЕРЖДАЮ

Ответственный за образовательную
программу

проф., д.т.н., доц.

(должность, уч. степень, звание)

С.В. Беззатеев

(инициалы, фамилия)

(подпись)

«27» июня 2024 г

Лист согласования рабочей программы дисциплины

Программу составил (а)

проф., к.т.н., проф.

(должность, уч. степень, звание)

27.06.2024

(подпись, дата)

С.Г. Фомичева

(инициалы, фамилия)

Программа одобрена на заседании кафедры № 33

«27» июня 2024 г, протокол № 11

Заведующий кафедрой № 33

д.т.н., доц.

(уч. степень, звание)

27.06.2024

(подпись, дата)

С.В. Беззатеев

(инициалы, фамилия)

Заместитель директора института №3 по методической работе

(должность, уч. степень, звание)

27.06.2024

(подпись, дата)

Н.В. Решетникова

(инициалы, фамилия)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Основы программирования»
(Наименование дисциплины)

Код направления подготовки/ специальности	10.05.03
Наименование направления подготовки/ специальности	Информационная безопасность автоматизированных систем
Наименование направленности	Безопасность открытых информационных систем
Форма обучения	очная
Год приема	2024

Санкт-Петербург– 2024

Аннотация

Дисциплина «Основы программирования» входит в образовательную программу высшего образования – программу специалитета по направлению подготовки/ специальности 10.05.03 «Информационная безопасность автоматизированных систем» направленности «Безопасность открытых информационных систем». Дисциплина реализуется кафедрой «№33».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ОПК-7 «Способен создавать программы на языках общего назначения, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ»

Содержание дисциплины охватывает круг вопросов, связанных с изучением основных особенностей разработки программ на одном из языков программирования высокого уровня, разработка и использование составных типов данных, разработка и создание динамических переменных, динамических структур данных, разработка программ с использованием современных инструментальных средств с использованием технологии быстрой разработки приложений (RAD). Формирование у будущих специалистов твердых теоретических знаний и практических навыков по разработке программ с использованием языка высокого уровня.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, практические занятия, самостоятельная работа студента, консультации, курсовое проектирование.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме экзамена.

Общая трудоемкость освоения дисциплины составляет 9 зачетных единиц, 324 часа.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Целью дисциплины является изучение основных особенностей разработки программ на одном из языков программирования высокого уровня, разработка и использование составных типов данных, разработка и создание динамических переменных, динамических структур данных, разработка программ с использованием современных инструментальных средств с использованием технологии быстрой разработки приложений (RAD). Формирование у будущих специалистов твердых теоретических знаний и практических навыков по разработке программ с использованием языка высокого уровня C++. Основной задачей изучения дисциплины является реализация требований, установленных в Государственном стандарте высшего профессионального образования к подготовке специалистов. Изучение методов и способов разработки алгоритмов и программ с использованием языка высокого уровня C++. В ходе изучения дисциплины ставятся следующие задачи: приобретение системы знаний о разработке программ как одной из функций деятельности информатика, позволяющей принять соответствующие решения при решении задач в экономической отрасли; организация разработки информационной системы для широкого круга внутренних и внешних пользователей предприятия; формирование и представление информации, использования её при разработке приложений, удовлетворяющей требованиям различных пользователей программного обеспечения.

1.2. Дисциплина входит в состав обязательной части образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

Категория (группа) компетенции	Код и наименование компетенции	Код и наименование индикатора достижения компетенции
Общепрофессиональные компетенции	ОПК-7 Способен создавать программы на языках общего назначения, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ	ОПК-7.3.1 знать основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий ОПК-7.У.1 уметь решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением средств и методов программирования и с учетом основных требований информационной безопасности ОПК-7.В.1 владеть навыками использования методов программирования и стандартных прикладных программ для решения профессиональных задач в области информационной безопасности и защиты информации

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- Информатика
- Учебная (ознакомительная) практика

Знания, полученные при изучении материала данной дисциплины, имеют как самостоятельное значение, так и используются при изучении других дисциплин:

- Технологии и методы программирования
- Программно-аппаратные средства обеспечения информационной безопасности
- Научно-исследовательская работа

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

Вид учебной работы	Всего	Трудоемкость по семестрам	
		№2	№3
1	2	3	4
Общая трудоемкость дисциплины, ЗЕ/ (час)	9/ 324	5/ 180	4/ 144
Из них часов практической подготовки			
Аудиторные занятия, всего час.	187	102	85
в том числе:			
лекции (Л), (час)	68	34	34
практические/семинарские занятия (ПЗ), (час)	34	34	
лабораторные работы (ЛР), (час)	68	34	34
курсовой проект (работа) (КП, КР), (час)	17		17
экзамен, (час)	90	54	36
Самостоятельная работа, всего (час)	47	24	23
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.**)	Экз., Экз.	Экз.	Экз.

4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий.

Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

Разделы, темы дисциплины	Лекции (час)	ПЗ (СЗ) (час)	ЛР (час)	КП (час)	СРС (час)
Семестр 2					
Раздел 1. Введение в информационные технологии. Освоение среды разработки. Разработка и отладка приложений линейной структуры.	8	8	8		2
Раздел 2. Техническое и программное обеспечение ЭВМ. Разработка и отладка приложений разветвляющейся и циклической структуры, обработка одномерных массивов	8	8	8		8

Раздел 3. Технические средства и программное обеспечение ЭВМ. Разработка и отладка приложений по обработке двумерных массивов с использованием подпрограмм.	8	8	8		8
Раздел 4. Компьютерные сети. Базы данных. Разработка и отладка приложений по обработке строковой информации	10	10	10		6
Итого в семестре:	34	34	34		24
Семестр 3					
Раздел 5. Разработка и отладка приложений с использованием структур, универсальных модулей и нескольких форм.	8		8		4
Раздел 6. Разработка и отладка приложений с использованием типизированных файлов.	8		8		4
Раздел 7. Разработка и отладка приложений с использованием текстовых файлов	8		8		4
Раздел 8. Графические возможности программирования	10		10		11
Выполнение курсовой работы				17	
Итого в семестре:	34		34	17	23
Итого	68	34	68	17	47

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

Номер раздела	Название и содержание разделов и тем лекционных занятий
Раздел 1. Введение в информационные технологии. Освоение среды разработки. Разработка и отладка приложений линейной структуры.	Введение. Информация и информатика. Основные задачи учебной дисциплины. Основные понятия: информация, информатизация, информационные технологии информатика. Алгебра логики. Системы счисления. История развития вычислительной техники. Вычислительная техника и научно-технический прогресс. Использование ЭВМ в научной, инженерной и экономических областях. Применение ЭВМ в интеллектуальных системах принятия решений и управления, в системах автоматизированного проектирования. Классификация ЭВМ.
Раздел 2. Техническое и программное обеспечение ЭВМ. Разработка и отладка приложений разветвляющейся и циклической структуры, обработка одномерных массивов	Обобщенная структурная схема ЭВМ. Процессор и оперативная память. Принцип автоматической обработки информации в ЭВМ. Основные технические характеристики ЭВМ. Внешние запоминающие устройства. Размещение информации на носителях. Устройства ввода-вывода информации. Персональные ЭВМ, их основные технические характеристики. Назначение, состав и структура программного обеспечения. Обработка программ под управлением операционной системы. Дружественный интерфейс. Драйверы. Сервисные средства. Пакеты прикладных программ. Общая характеристика языков программирования, области их применения. Компиляторы и интерпретаторы. Системы программирования. Технологии разработки программ. Основы структурного

	программирования. Базовые управляющие конструкции. Тестирование и отладка программ.
Раздел 3. Технические средства и программное обеспечение ЭВМ. Разработка и отладка приложений по обработке двумерных массивов с использованием подпрограмм.	Вычислительные комплексы и сети. Локальные сети. Структура вычислительных сетей. Виды топологии сети. Глобальная сеть. Сетевые протоколы. Доменные имена. Основные сервисы глобальной сети. Базы данных. Типы баз данных. Структура базы данных. Требования к базам данных. Реляционные модели данных. Типы отношений. Нормализация отношений.
Раздел 4. Компьютерные сети. Базы данных. Разработка и отладка приложений по обработке строковой информации	Взаимодействие пользователя с базой данных. Системы управления базами данных (СУБД). Основные функции СУБД. Знакомство с основными алгоритмами обработки информации, их анализ и сравнение.
Раздел 5. Разработка и отладка приложений с использованием структур, универсальных модулей и нескольких форм.	Тип данных – структура. Правила работы со структурами, их полями и методами Понятие универсального модуля. Усвоение целесообразности использования модулей при программировании сложных задач Создание модулей, содержащих подпрограммы обработки массивов структур. Работа с несколькими экранными формами в приложении к задачам обработки массивов структур
Раздел 6. Разработка и отладка приложений с использованием типизированных файлов.	Знакомство с файлами и основными функциями, и процедурами их обработки. Особенности файлов прямого доступа (типизированных). Обработка типизированных файлов с помощью подпрограмм.
Раздел 7. Разработка и отладка приложений с использованием текстовых файлов	Особенности файлов последовательного доступа (текстовых) Обработка текстовых файлов с использованием меню
Раздел 8. Графические возможности программирования	Знакомство с графическими возможностями среды программирования. Компоненты среды, а также процедуры и функции для изображения графических примитивов Создание рисунков. Преобразование и анимация изображений

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

№ п/п	Темы практических занятий	Формы практических занятий	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Семестр 2					
1	Знакомство со средой программирования: главное окно проекта, экранная форма, элементы управления и их свойства, главное меню, окно кода программы, работа с редактором, использование справочной службы	Обсуждение практических примеров	2		1

2	Знакомство с основами языка программирования (типы данных, структура программы, операторы, выражения, библиотечные функции)	Обсуждение практических примеров	2		1
3	Процедуры ввода, вывода и оператор присваивания. Организация приложений линейной структуры. Анализ возможных ошибок, разработка набора текстовых данных и использование программы-отладчика среды разработки	Обсуждение практических примеров	2		1
4	Осваивание техники проведения процедуры отладки (точки контрольного останова, окно наблюдения, принудительное прерывание работы приложения, трассировка, действия в точках прерывания, вычисление выражений и изменение значений)	Обсуждение практических примеров	2		1
5	Логические операции. Оператор перехода. Условный оператор. Функция условного перехода. Оператор выбора. Организация приложений разветвляющейся структуры. Операторы цикла с известным числом повторений.	Обсуждение практических примеров	2		2

6	Операторы цикла с неизвестным числом повторений. Осваивание циклических структур и типовые алгоритмы: накопление суммы, произведения, вычисление факториала на примерах с последовательностями с неизвестным количеством элементов.	Обсуждение практических примеров	2		2
7	Обработка одномерных числовых массивов (ввод, вывод, создание, изменение). Нахождение суммы и произведения. Алгоритмы сортировки массивов (метод обмена, метод вставки и т.п.), поиска в массиве (линейный и бинарный поиск). Сравнение алгоритмов. Нахождение минимального и максимального элементов массива	Обсуждение практических примеров	2		2
8	Знакомство с подпрограммами. Структура программы с подпрограммой. Параметры подпрограмм. Освоение подпрограмм – функций	Обсуждение практических примеров	2		2
9	Двумерные числовые массивы. Понятие двумерных динамических массивов	Обсуждение практических примеров	2		3
10	Ввод – вывод матриц с использованием процедур. Алгоритмы преобразования матриц	Обсуждение практических примеров	2		3

11	Нахождение минимального и максимального элементов в строке (столбце) матрицы с использованием подпрограмм.	Обсуждение практических примеров	2		3
12	Квадратные матрицы. понятие главной и побочной диагоналей. Алгоритмы обработки квадратных матриц.	Обсуждение практических примеров	2		3
13	Обработка строк. Функции и процедуры для работы со строками	Обсуждение практических примеров	2		4
14	Работа с многострочным текстом	Обсуждение практических примеров	2		4
15	Разбиение строки на слова. Выделение чисел из строки	Обсуждение практических примеров	2		4
16	Работа с многострочным текстом с использованием подпрограмм	Обсуждение практических примеров	2		4
Всего			34		

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

№ п/п	Наименование лабораторных работ	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Семестр 2				
1	Освоение среды разработки Приложения линейной структуры Осваивание техники проведения процесса отладки	4		1
2	Организация приложений разветвляющейся структуры	4		1
3	Циклы с известным числом повторений Циклы с неизвестным числом повторений Одномерные числовые массивы. Ввод, вывод, нахождение суммы, произведения элементов массива. Одномерные числовые массивы. Нахождение минимального и максимального элементов массива.	4		2
4	Одномерные динамические массивы. Методы сортировки.	4		2

5	Обработка двумерных массивов. Использование процедур Обработка двумерных массивов. Использование функций	2		3
6	Квадратные матрицы Обработка квадратных матриц с использованием функций	2		3
7	Строковый тип данных Типовые приемы обработки строк: поиск, копирование, удаление и вставка фрагментов строки	4		4
7	Понятие универсального модуля Разработка и отладка модулей с использованием подпрограмм.	4		4
9	Многомодульные приложения.	2		4
Семестр 3				
5	Пользовательский тип данных. Тип данных – структура. Отработка основных приемов программирования на примерах задач с массивами структур.	4		5
6	Конструирование сложных структур, включающих различные методы обработки своих полей. Использование нескольких экранных форм.	4		5
7	Типизированные файлы. Приложения с меню	4		6
8	Типизированные файлы, компонентами которых являются структуры	4		6
9	Текстовые файлы Использование меню при разработке приложений с текстовыми файлами и типизированными файлами	4		7
10	Знакомство с графическими примитивами, с заданием цвета Построение неподвижных изображений	4		7
11	Анимация Построение движущихся изображений	4		8
12	Поворот и вращение изображений Масштабирование изображений	4		8
13	Комбинированное движение	2		8
Всего		68		

4.5. Курсовое проектирование/ выполнение курсовой работы

Цель курсовой работы:

Примерные темы заданий на курсовую работу приведены в разделе 10 РПД.

4.6. Самостоятельная работа обучающихся

Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

Вид самостоятельной работы	Всего, час	Семестр 2, час	Семестр 3, час
1	2	3	4
Изучение теоретического материала дисциплины (ТО)	20	10	10
Курсовое проектирование (КП, КР)			
Расчетно-графические задания (РГЗ)			
Выполнение реферата (Р)			
Подготовка к текущему контролю успеваемости (ТКУ)	9	6	3
Домашнее задание (ДЗ)			
Контрольные работы заочников (КРЗ)			
Подготовка к промежуточной аттестации (ПА)	18	8	10
Всего:	47	24	23

5. Перечень учебно-методического обеспечения
для самостоятельной работы обучающихся по дисциплине (модулю)

Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. 7-11.

6. Перечень печатных и электронных учебных изданий

Перечень печатных и электронных учебных изданий приведен в таблице 8.

Таблица 8– Перечень печатных и электронных учебных изданий

Шифр/ URL адрес	Библиографическая ссылка	Количество экземпляров в библиотеке (кроме электронных экземпляров)
004.4 Т 33	Теория языков программирования и методы трансляции [Текст] : методические указания к выполнению курсовой работы / С.-Петербург. гос. ун-т аэрокосм. приборостроения ; сост. Т. М. Максимова. - СПб. : Изд-во ГУАП, 2011. - 39 с. : табл. - Библиогр.: с. 26 (5 назв.). - Б. ц.	130
004.4 Б 46	Бенкен, Е. PHP, MySQL, XML программирование для Интернета [Текст] / Е. Бенкен. - 3-е изд. - СПб. : БХВ - Петербург, 2011. - 304 с. : рис. + 1 эл. опт. диск (CD-ROM). - Предм. указ.: с. 285 - 288. - ISBN 978-5-9775-0724-0 : 328.00 р.	6
004.4 И 21	Иванова, Г. С. Технология программирования [Текст] : учебник / Г. С. Иванова. - М. : КноРус, 2011. - 333 с. : рис., табл. - Библиогр.: с. 329 - 331. - Предм. указ.: с. 332 - 333. - ISBN 978-5-406-00519-4 : 297.00 р. Издание имеет гриф УМО по университетскому политехническому образованию.	22
004 В 52	Вирт, Н. Алгоритмы и структуры данных. Новая версия для Оберона + CD [Текст] / Н. Вирт ; пер. Д. Б. Подшивалов. - 2-е изд., испр. - М. : ДМК Пресс, 2012. - 272 с. : табл. + 1 эл.	10

	опт. диск (CD-ROM). - (Классика программирования). - Библиогр. в конце глав. - ISBN 978-5-94074-734-5 (рус.). - ISBN0-13-022005-9 (англ.) : 289.00 р. Приложение: Алгоритмы и структуры данных. - М. : ДМК Пресс, 2012. - 1. Шифр -00742	
004 С 30	Семакин, И. Г. Основы алгоритмизации и программирования [Текст] : учебник для СПО / И. Г. Семакин, А. П. Шестаков. - 3-е изд. стер. - М. : Академия, 2012. - 391 с. : рис. - (Среднее профессиональное образование. Информатика и вычислительная техника). - ISBN 978-5-7695-8957-7 : 624.80 р. Имеет гриф Минобрнауки РФ	7
004.4 Г 95	Гуриков, С. Р. Введение в программирование на языке Visual C# [Текст] : учебное пособие / С. Р. Гуриков. - М. : ФОРУМ : ИНФРА-М, 2013. - 448 с. : рис. - (Высшее образование: Бакалавриат). - Библиогр.: с. 446 (4 назв.). - ISBN 978-5-91134-738-3 (ФОРУМ). - ISBN978-5-16-006674-5 (ИНФРА-М) : 636.90 р. На с. 272 -445: Приложения: Примеры заполнения лабораторных работ. Создание заставок к проектами	10
004.4 Э 38	Эккель, Брюс. Философия Java [Текст] / Б. Эккель. - 4-е изд. - СПб. : ПИТЕР, 2011. - 637 с. - (Библиотека программиста). - Алф. указ.: с. 631 - 637. - ISBN 978-5-388-00003-3 : 650.00 р.	1

7. Перечень электронных образовательных ресурсов

информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

URL адрес	Наименование
www.intuit.ru/	Национальный Открытый Университет "ИНТУИТ"

8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

№ п/п	Наименование
	Не предусмотрено

8.2. Перечень информационно-справочных систем, используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

№ п/п	Наименование
	Не предусмотрено

9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице 12.

Таблица 12 – Состав материально-технической базы

№ п/п	Наименование составной части материально-технической базы	Номер аудитории (при необходимости)
1	Лекционная аудитория	
2	Компьютерная лаборатория	

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

Вид промежуточной аттестации	Перечень оценочных средств
Экзамен	Список вопросов к экзамену; Экзаменационные билеты; Задачи; Тесты.
Выполнение курсовой работы	Экспертная оценка на основе требований к содержанию курсовой работы по дисциплине.

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 –Критерии оценки уровня сформированности компетенций

Оценка компетенции	Характеристика сформированных компетенций
5-балльная шкала	
«отлично» «зачтено»	– обучающийся глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно привязывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения;

Оценка компетенции	Характеристика сформированных компетенций
5-балльная шкала	
	– свободно владеет системой специализированных понятий.
«хорошо» «зачтено»	– обучающийся твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы; – не допускает существенных неточностей; – увязывает усвоенные знания с практической деятельностью направления; – аргументирует научные положения; – делает выводы и обобщения; – владеет системой специализированных понятий.
«удовлетворительно» «зачтено»	– обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы; – допускает несущественные ошибки и неточности; – испытывает затруднения в практическом применении знаний направления; – слабо аргументирует научные положения; – затрудняется в формулировании выводов и обобщений; – частично владеет системой специализированных понятий.
«неудовлетворительно» «не зачтено»	– обучающийся не усвоил значительной части программного материала; – допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении; – испытывает трудности в практическом применении знаний; – не может аргументировать научные положения; – не формулирует выводов и обобщений.

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

№ п/п	Перечень вопросов (задач) для экзамена	Код индикатора
	1 Понятие указателя. Синтаксис описания указателя. 2 Инициализация указателей. 3 Операции с указателями. Адресная арифметика. 4 Понятие ссылки. Синтаксис описания. 5 Структура как тип данных. Синтаксис описания типа. 6 Инициализация структур. 7 Односвязный список. Добавление элемента в начало списка. 8 Односвязный список. Добавление элемента в середину списка. 9 Односвязный список. Удаление элемента из начала списка. 10 Односвязный список. Удаление элемента из середины списка. 11 Понятие стека. Занесение и выборка элементов из вершины стека. 12 Изменение и чтение элемента без его выборки из стека.	ОПК-7.3.1
	13 Понятие очереди. Организация очередей.	ОПК-7.У.1

	<p>14 Инициализация очереди. 15 Выборка элемента из очереди и проверка очереди на пустоту. 16 Добавление элемента в конец очереди. 17 Понятие динамического массива. 18 Инициализация динамических массивов, размещение их в динамической области памяти. 19 Использование массивов в качестве параметров функции. 20 Указатели на функции и их использование. 21 Понятие файлов в C++. 22 Поточковый ввод-вывод, основные понятия. 23 Открытие, закрытие потокового ввода-вывода в файл. 24 Дополнительные функции для работы с потоковым вводом-выводом в файл. 25 Режимы файлов. 26 Понятие двоичных файлов. 27 Работа с двоичными файлами. 28 Произвольный доступ к файлам. 29 Интегрированная Среда Разработки (Integrated Development Environment (IDE) C++Builder 6. Основные элементы.</p>	
	<p>30 Состав инструментальных панелей быстрых кнопок C++ Builder. 31 Понятие компонентов C++Builder . Разновидности компонентов. 32 Свойства компонентов. Понятие и их назначение. 33 События, связанные с компонентами. Понятие и их назначение. 34 Последовательность создания консольного приложения в C++Builder. 35 Разработка простейшего приложения Windows в C++Builder. 36 Форма (объект тип TForm). Основные свойства формы. 37 Свойства компонента Label (поле вывода текста). 38 Свойства компонента Edit (поле редактирования). 39 Свойства компонента Button (командная кнопка). 40 Свойства компонента Memo. 41 Свойства компонента RadioButton. 42 Свойства компонента CheckBox. 43 Свойства компонента ListBox. 44 Свойства компонента ComboBox. 45 Свойства компонента StringGrid. 46 Свойства компонента image. 47 Свойства компонента Timer. 48 Разновидности событий.</p>	ОПК-7.В.1

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.
Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

№ п/п	Перечень вопросов (задач) для зачета / дифф. зачета	Код индикатора
	Учебным планом не предусмотрено	

Перечень тем для курсового проектирования/выполнения курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для курсового проектирования/выполнения курсовой работы

№ п/п	Примерный перечень тем для курсового проектирования/выполнения курсовой работы
	<ol style="list-style-type: none"> 1. Разработка приложений с использованием методов численного интегрирования. 2. Разработка приложений с использованием методов численного дифференцирования. 3. Разработка приложений с решением задач Коши методом Эйлера. 4. Разработка приложений с решением задач Коши методом Рунге-Кутты. 5. Разработка приложений с решением задач Коши методом прогноза и коррекции. 6. Разработка приложений с решением системы линейных уравнений методом Гаусса. 7. Разработка приложений с решением нелинейных и трансцендентных уравнений методами простых итераций, Ньютона (касательных), дихотомии (метод деления отрезка пополам). 8. Разработка приложения одномерной минимизации функции методом золотого сечения. 9. Разработка приложения минимизации функции методом градиентного спуска. 10. Разработка графических приложений. 11. Разработка простейших баз данных. 12. Разработка файловых менеджеров. 13. Разработка приложений с использованием графики. 14. Разработка компьютерных игр.

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

№ п/п	Примерный перечень вопросов для тестов	Код индикатора
	<p>Какие среды программирования (IDE) предназначены для разработки программных средств?</p> <ul style="list-style-type: none"> ○ MVS, NetBeans, QT Creator, RAD Studio, Dev-C++ ○ MVS, Code::Blocks, QT Creator, AutoCAD, Eclipse ○ MVS, Code::Blocks, QT Creator, RAD Studio, MathCAD <p>Структура объявления переменных в C++</p> <ul style="list-style-type: none"> ○ [=], <идент. 2>, ...; ○ [=]; <идент. 2>, ...; ○ [:=], <идент. 2>, ...; ○ [==]; <идент. 2>, ...; <p>4. Какой из следующих операторов - оператор сравнения двух переменных?</p> <ul style="list-style-type: none"> ○ == ○ equal ○ := ○ = <p>5. Какой из ниже перечисленных операторов, не является циклом в</p>	

	<p>C++?</p> <ul style="list-style-type: none"> ○ for ○ while ○ repeat until ○ do while <p>6. Какой из перечисленных типов данных не является типом данных в C++?</p> <ul style="list-style-type: none"> ○ float ○ real ○ int ○ double <p>7. Программа, переводящая входную программу на исходном языке в эквивалентную ей выходную программу на результирующем языке, называется:</p> <ul style="list-style-type: none"> ○ интерпретатор ○ транслятор ○ компилятор ○ сканер <p>8. Какой служебный знак ставится после оператора case ?</p> <ul style="list-style-type: none"> ○ ; ○ . ○ - ○ : <p>9. Чему будет равна переменная a, после выполнения этого кода <code>int a; for(a = 0; a < 10; a++) {}</code>?</p> <ul style="list-style-type: none"> ○ 1 ○ 9 ○ 10 <p>10. Название C++ предложил</p> <ul style="list-style-type: none"> ○ Рик Масситти ○ Дональд Кнут ○ Кэн Томпсон ○ Бьерн Страуструп 	
--	--	--

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

№ п/п	Перечень контрольных работ
	Не предусмотрено

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру

проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

1. Методические указания для обучающихся по освоению дисциплины

Целью дисциплины является изучение основных особенностей разработки программ на одном из языков программирования высокого уровня, разработка и использование составных типов данных, разработка и создание динамических переменных, динамических структур данных, разработка программ с использованием современных инструментальных средств с использованием технологии быстрой разработки приложений (RAD). Формирование у будущих специалистов твердых теоретических знаний и практических навыков по разработке программ с использованием языка высокого уровня C++. Основной задачей изучения дисциплины является реализация требований, установленных в Государственном стандарте высшего профессионального образования к подготовке специалистов. Изучение методов и способов разработки алгоритмов и программ с использованием языка высокого уровня C++. В ходе изучения дисциплины ставятся следующие задачи: приобретение системы знаний о разработке программ как одной из функций деятельности информатика, позволяющей принять соответствующие решения при решении задач в экономической отрасли; организация разработки информационной системы для широкого круга внутренних и внешних пользователей предприятия; формирование и представление информации, использования её при разработке приложений, удовлетворяющей требованиям различных пользователей программного обеспечения.

Методические указания для обучающихся по освоению лекционного материала

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимися лекционного материала:

- получение современных, целостных, взаимосвязанных знаний, уровень которых определяется целевой установкой к каждой конкретной теме;
- получение опыта творческой работы совместно с преподавателем;
- развитие профессионально–деловых качеств, любви к предмету и самостоятельного творческого мышления.
- появление необходимого интереса, необходимого для самостоятельной работы;
- получение знаний о современном уровне развития науки и техники и о прогнозе их развития на ближайшие годы;
- научиться методически обрабатывать материал (выделять главные мысли и положения, приходиться к конкретным выводам, повторять их в различных формулировках);
- получение точного понимания всех необходимых терминов и понятий.

Лекционный материал может сопровождаться демонстрацией слайдов и использованием раздаточного материала при проведении коротких дискуссий об особенностях применения отдельных тематик по дисциплине.

Структура предоставления лекционного материала:

- Раздел 1. Технология создания алгоритмов программ
- Раздел 2. Программирование обработки текстовой информации

- Раздел 3. Сложные типы данных
 - Раздел 4. Основы конструирования программ методом функциональной декомпозиции.
- Модульное программирование
- Раздел 5. Разработка приложений работы с файлами и файловой структурой
 - Раздел 6. Разработка приложений с использованием динамических структур данных
 - Раздел 7. Рекурсивные структуры данных и рекурсивные алгоритмы обработки
 - Раздел 8. Разработка консольных приложений с использованием стандартного модуля `conio.h`
 - Раздел 9. Разработка графических приложений с использованием стандартного графического модуля
 - Раздел 10. Основы разработки приложений с использованием инструментальной среды разработки и отладки программ C++ Builder.

Методические указания для обучающихся по прохождению практических занятий

Практическое занятие является одной из основных форм организации учебного процесса, заключающейся в выполнении обучающимися под руководством преподавателя комплекса учебных заданий с целью усвоения научно-теоретических основ учебной дисциплины, приобретения умений и навыков, опыта творческой деятельности.

Целью практического занятия для обучающегося является привитие обучающемуся умений и навыков практической деятельности по изучаемой дисциплине.

Планируемые результаты при освоении обучающимся практических занятий:

- закрепление, углубление, расширение и детализация знаний при решении конкретных задач;
- развитие познавательных способностей, самостоятельности мышления, творческой активности;
- овладение новыми методами и методиками изучения конкретной учебной дисциплины;
- выработка способности логического осмысления полученных знаний для выполнения заданий;
- обеспечение рационального сочетания коллективной и индивидуальной форм обучения.

Функции практических занятий:

- познавательная;
- развивающая;
- воспитательная.

По характеру выполняемых обучающимся заданий по практическим занятиям подразделяются на:

- ознакомительные, проводимые с целью закрепления и конкретизации изученного теоретического материала;
- аналитические, ставящие своей целью получение новой информации на основе формализованных методов;
- творческие, связанные с получением новой информации путем самостоятельно выбранных подходов к решению задач.

Формы организации практических занятий определяются в соответствии со специфическими особенностями учебной дисциплины и целями обучения. Они могут проводиться:

- в интерактивной форме (решение ситуационных задач, занятия по моделированию реальных условий, деловые игры, игровое проектирование, имитационные занятия, выездные занятия в организации (предприятия), деловая учебная игра, ролевая игра, психологический тренинг, кейс, мозговой штурм, групповые дискуссии);

– в не интерактивной форме (выполнение упражнений, решение типовых задач, решение ситуационных задач и другое).

Методика проведения практического занятия может быть различной, при этом важно достижение общей цели дисциплины.

Требования к проведению практических занятий

На практических занятиях под руководством преподавателя, решают практические задачи.

При проведении практических занятиях применяются следующие интерактивные методы обучения:

Дискуссия - форма учебной работы, в рамках которой студенты высказывают свое мнение по проблеме, заданной преподавателем. Проведение дискуссий по проблемным вопросам подразумевает написание студентами эссе, тезисов или рефератов по предложенной тематике. Дискуссия групповая - метод организации совместной коллективной деятельности, позволяющий в процессе непосредственного общения путем логических доводов воздействовать на мнения, позиции и установки участников дискуссии. Целью дискуссии является интенсивное и продуктивное решение групповой задачи. Метод групповой дискуссии обеспечивает глубокую проработку имеющейся информации, возможность высказывания студентами разных точек зрения по заданной преподавателем проблеме, тем самым, способствуя выработке адекватного в данной ситуации решения. Метод групповой дискуссии увеличивает вовлеченность участников в процесс этого решения, что повышает вероятность его реализации.

Доклад (презентация) - публичное сообщение, представляющие собой развернутое изложение определенной темы, вопроса программы. Доклад может быть представлен различными участниками процесса обучения: преподавателем, приглашенным экспертом, студентом, группой студентов. Доклады направлены на более глубокое изучение студентами лекционного материала или рассмотрения вопросов для дополнительного изучения.

Интерактивные методы обучения - методы обучения, при которых сам процесс передачи информации построен на принципе активного двухстороннего взаимодействия преподавателя и студента. Он предполагает большую активность студента, его творческое переосмысление полученных сведений. Основные критерии интерактивной модели обучения: возможность неформальной дискуссии, свободного изложения материала, наличие групповых заданий, которые требуют коллективных усилий, инициативность студента, постоянный контроль во время семестра, выполнение письменных работ. Интерактивные методы включают: метод презентации, дискуссии, метод текущего контроля, метод тестирования и др.

Исследовательский метод обучения - организация обучения на основе поисковой, познавательной деятельности студентов путем постановки преподавателем познавательных и практических задач, требующих самостоятельного творческого решения. Сущность исследовательского метода обучения обусловлена его функциями. Метод организует творческий поиск и применение знаний, является условием формирования интереса, потребности в творческой деятельности, в самообразовании. Основная идея исследовательского метода обучения заключается в использовании научного подхода к решению той или иной учебной задачи. Работа студентов в этом случае строится по логике проведения классического научного исследования с использованием всех научно-исследовательских методов и приемов, характерных для деятельности ученых. Основные этапы организации учебной деятельности при использовании исследовательского метода: 1.Определение общей темы исследования, предмета и объекта исследования. 2.Выявление и формулирование общей проблемы. 3.Формулировка гипотез. 4.Определение методов сбора и обработки данных в подтверждение выдвинутых гипотез. 5.Сбор данных. 6.Обсуждение полученных данных.

7.Проверка гипотез. 8.Формулировка понятий, обобщений, выводов. 9.Применение заключений, выводов.

Пост-тест - тест на оценку, позволяющий проверить знания студентов по пройденным темам. Данный метод обучения используется в учебном процессе при проведении тестирования с использованием аттестационного педагогического измерительного материала для оценки качества знаний студентов по дисциплине.

Методические указания для обучающихся по прохождению лабораторных работ

В ходе выполнения лабораторных работ обучающийся должен углубить и закрепить знания, практические навыки, овладеть современной методикой и техникой эксперимента в соответствии с квалификационной характеристикой обучающегося. Выполнение лабораторных работ состоит из экспериментально-практической, расчетно-аналитической частей и контрольных мероприятий.

Выполнение лабораторных работ обучающимся является неотъемлемой частью изучения дисциплины, определяемой учебным планом, и относится к средствам, обеспечивающим решение следующих основных задач у обучающегося:

- приобретение навыков исследования процессов, явлений и объектов, изучаемых в рамках данной дисциплины;
- закрепление, развитие и детализация теоретических знаний, полученных на лекциях;
- получение новой информации по изучаемой дисциплине;
- приобретение навыков самостоятельной работы с лабораторным оборудованием и приборами.

Задание и требования к проведению лабораторных работ

Цель проведения лабораторных занятий заключается в закреплении полученных теоретических знаний на лекциях и в процессе самостоятельного изучения студентами специальной литературы. Основной формой проведения лабораторных занятий является обсуждение наиболее проблемных и сложных ситуаций по отдельным темам, а также выполнение различных заданий с использованием компьютерных технологий. В обязанности преподавателя входят оказание методической помощи и консультирование студентов по соответствующим темам курса.

Примерные задания к лабораторным работам:

1. Разработать приложение, использующее строковые типы данных
2. Разработать приложение с использованием структур
3. Разработать приложение с использованием динамических переменных – использование указателей для работы с динамическими переменными; – использование динамической структуры данных – стек; – использование динамической структуры данных – список; использование динамической структуры данных – очередь
4. Разработать приложение с использованием динамических массивов
5. Разработать приложение с использованием файлового ввода-вывода
6. Разработать приложение в среде Builder: – разработать программ с использованием инструментария; – структура проекта; – использование компонент
7. Программирование с использованием компонент работы со строками – типы данных для работы со строками; – используемые компоненты для работы со строками; обработка событий.
8. Разработать приложение с использованием диалогов для выбора файлов и главного меню – компоненты диалога с пользователем; – приёмы программирования со структурами; – приёмы программирования с файлами.
9. Разработать приложение с использованием средств для отображения графической информации
10. Разработать приложение с использованием доступа к таблицам баз данных – Основы реляционных баз данных; – Создание таблиц при помощи DataBase Desktop; – Разработать приложения для работы с таблицей данных; – Обработка информации однотабличной базы данных; – Печать данных.

Структура и форма отчета о лабораторной работе

Отчёт по лабораторной работе оформляется индивидуально каждым студентом, выполнившим необходимые (независимо от того, выполнялся ли эксперимент индивидуально или в составе группы студентов). Страницы отчёта следует пронумеровать (титульный лист не нумеруется, далее идет страница 2 и т.д.). Титульный лист отчёта должен содержать фразу: «Отчёт по лабораторной работе «Название работы», чуть ниже: Выполнил студент группы (номер группы) (Фамилия, инициалы)». Внизу листа следует указать текущий год. Например, Отчёт по лабораторной работе № (номер работы) «Введение в спектральный анализ», Выполнил студент группы 5221 Иванов И.И. Вторая страница текста, следующая за титульным листом, должна начинаться с пункта: Цель работы. Отчёт, как правило, должен содержать следующие основные разделы:

1. Цель работы;
2. Теоретическая часть;
3. Программное обеспечение, используемое в работе;
4. Результаты;
5. Выводы.

В случае необходимости в конце отчёта приводится перечень литературы.

Требования к оформлению отчета о лабораторной работе

Теоретическая часть должна содержать минимум необходимых теоретических сведений о предметной области. Не следует копировать целиком или частично методическое пособие (описание) лабораторной работы или разделы учебника.

В разделе Программное обеспечение необходимо описать, с помощью каких инструментальных средств и каким образом были разработаны модели и получены результаты. Рисунки, блок-схемы, описание модели и её особенностей, необходимость отладки – все это должно быть представлено в указанном разделе.

Раздел Результаты включает в себя скриншоты программного приложения, полученные при выполнении лабораторной работы. Рисунки, графики и таблицы нумеруются и подписываются заголовками.

Выводы не должны быть простым перечислением того, что сделано. Здесь важно отметить, какие новые знания о предмете исследования были получены при выполнении работы, к чему привело обсуждение результатов, насколько выполнена заявленная цель работы. Выводы по работе каждый студент делает самостоятельно. В случае необходимости в конце отчёта приводится Список литературы, использованной при подготовке к работе. В тексте отчёта делаются краткие ссылки на литературу (учебники, справочники, иные источники...) номером в квадратных скобках, напр., [1]. Литературные источники нумеруются по мере их появления в тексте отчёта. В конце отчёта даётся их подробный список. На все источники списка литературы должны быть ссылки в тексте отчёта, там, где это необходимо.

При сдаче отчёта преподаватель может сделать устные и письменные замечания, задать дополнительные вопросы. Все ответы на дополнительные вопросы, обсуждения выполняются студентом на отдельных листах, включаемых в отчёт (при этом в тексте основного отчёта делается сноска или другой значок, которому будет соответствовать новый материал). При этом письменные замечания преподавателя должны остаться в тексте для ясности динамики работы над отчётом.

Объём отчёта должен быть оптимальным для понимания того, что и как сделал студент, выполняя работу. Обязательные требования к отчёту включают общую и специальную грамотность изложения, а также аккуратность оформления.

После приёма преподавателем отчёт хранится на кафедре.

Методические указания для обучающихся по прохождению курсового проектирования/ работы

Курсовой проект/ работа проводится с целью формирования у обучающихся опыта комплексного решения конкретных задач профессиональной деятельности.

Курсовой проект/ работа позволяет обучающемуся:

- систематизировать и закрепить полученные теоретические знания и практические умения по профессиональным учебным дисциплинам и модулям в соответствии с требованиями к уровню подготовки, установленными программой учебной дисциплины, программой подготовки специалиста соответствующего уровня, квалификации;
- применить полученные знания, умения и практический опыт при решении комплексных задач, в соответствии с основными видами профессиональной деятельности по направлению/ специальности/ программе;
- углубить теоретические знания в соответствии с заданной темой;
- сформировать умения применять теоретические знания при решении нестандартных задач;
- приобрести опыт аналитической, расчётной, конструкторской работы и сформировать соответствующие умения;
- сформировать умения работы со специальной литературой, справочной, нормативной и правовой документацией и иными информационными источниками;
- сформировать умения формулировать логически обоснованные выводы, предложения и рекомендации по результатам выполнения работы;
- развить профессиональную письменную и устную речь обучающегося;
- развить системное мышление, творческую инициативу, самостоятельность, организованность и ответственность за принимаемые решения;
- сформировать навыки планомерной регулярной работы над решением поставленных задач.

Структура пояснительной записки курсовой работы / проекта

В соответствии с учебным планом студенты готовят курсовую работу. Цель выполнения курсовой работы: изучить методику разработки приложений с использованием современных инструментальных средств. Состав курсовой работы: Первая часть курсовой работы (электронная). В ней студент должен раскрыть практические знания о современных подходах разработки программ с использованием языка программирования C++. Представить исходный код самостоятельно разработанной программы, скомпилированный и отлаженный разработанный программный продукт на электронном носителе информации. Вторая часть (теоретическая). Приводится обзор нормативных актов по изучаемой теме, краткая теория и критический обзор литературных источников по проблеме исследования. Описать алгоритм, исходный код, методы программирования и библиотечные компоненты, использованные при разработке приложения. Привести описание работы приложения, результаты работы. При выполнении курсовой работы студент должен использовать язык программирования высокого уровня интегрированные среды разработки и отладки программ Visual C++ , Builder C++.

Основные разделы курсовой работы:

1. Титульный лист.
2. Утвержденное индивидуальное задание на курсовую работу.
3. Схема функциональной структуры программного приложения.
4. Описание автоматизируемых функций
5. Структурная схема иерархии форм приложения
6. Описание программного обеспечения
7. Программа и методика испытаний приложения
8. Основные результаты работы приложения.

Требования к оформлению пояснительной записки курсовой работы / проекта

В виду принадлежности курсового проекта к дисциплинам связанным с информационными технологиями и электронно-вычислительными машинами пояснительная записка должна быть оформлена при помощи любого программного инструмента и распечатана на листах формата А4

(210×297 мм), листы должны быть пронумерованы и сшиты. Поля листа должны составлять левое 25 мм, верхнее и нижнее 20 мм, правое 15 мм. Текст записки должен быть набран удобочитаемым шрифтом по размеру и начертанию соответствующий «Times New Roman» в 14 пт. Межстрочный интервал должен соответствовать полutorному. В записке также должен быть предусмотрен карман для помещения в него диска с работоспособным приложением и всеми исходными текстами программ. Допускается помещать на дискету архив в формате zip или rar.

Полный листинг программы должен включать в себя распечатку всех файлов программ, из которых состоит проект. Формы проекта должны быть распечатаны в двух видах: в виде формы и в виде тестового файла. Все файлы форм должны быть сгруппированы в следующей последовательности: сначала форма в процессе разработки, затем форма в текстовом виде и в завершении текст модуля связанный с формой. В записке фрагменты текстов программы, а также тексты распечаток модуля и формы должны быть выполнены шрифтом «Courier New» размером 10 пт., через одинарный интервал.

Титульный лист записки должен быть оформлен в соответствии с образцом, приведенным в приложении №1.

Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

Для обучающихся по заочной форме обучения, самостоятельная работа может включать в себя контрольную работу.

В процессе выполнения самостоятельной работы, у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет им развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическими материалами, направляющими самостоятельную работу обучающихся являются:

- учебно-методический материал по дисциплине;
- методические указания по выполнению контрольных работ (для обучающихся по заочной форме обучения).

Методические указания для обучающихся по прохождению промежуточной аттестации

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

– экзамен – форма оценки знаний, полученных обучающимся в процессе изучения всей дисциплины или ее части, навыков самостоятельной работы, способности применять их для решения практических задач. Экзамен, как правило, проводится в период экзаменационной сессии и завершается аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

– зачет – это форма оценки знаний, полученных обучающимся в ходе изучения учебной дисциплины в целом или промежуточная (по окончании семестра) оценка знаний обучающимся по отдельным разделам дисциплины с аттестационной оценкой «зачтено» или «не зачтено».

– дифференцированный зачет – это форма оценки знаний, полученных обучающимся при изучении дисциплины, при выполнении курсовых проектов, курсовых работ, научно-исследовательских работ и прохождении практик с аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Система оценок при проведении промежуточной аттестации осуществляется в соответствии с требованиями Положений «О текущем контроле успеваемости и промежуточной аттестации студентов ГУАП, обучающихся по программам высшего образования» и «О модульно-рейтинговой системе оценки качества учебной работы студентов в ГУАП».

Лист внесения изменений в рабочую программу дисциплины

Дата внесения изменений и дополнений. Подпись внесшего изменения	Содержание изменений и дополнений	Дата и № протокола заседания кафедры	Подпись зав. кафедрой