

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

Кафедра № 14

УТВЕРЖДАЮ

Руководитель направления

доц., к.т.н., доц.

(должность, уч. степень, звание)

Н.А. Овчинникова

(инициалы, фамилия)

(подпись)

«29» мая 2023 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Алгоритмизация и программирование»  
(Наименование дисциплины)

Код направления подготовки/ специальности	25.03.01
Наименование направления подготовки/ специальности	Техническая эксплуатация летательных аппаратов и двигателей
Наименование направленности	Эксплуатация и испытания авиационной и космической техники
Форма обучения	очная

Лист согласования рабочей программы дисциплины

Программу составил (а)

доц., к.т.н., доц.

(должность, уч. степень, звание)

(подпись, дата)

Е.П. Овсянников

(инициалы, фамилия)

Программа одобрена на заседании кафедры № 14

«25» мая 2023 г, протокол № 10

Заведующий кафедрой № 14

к.т.н., доц.

(уч. степень, звание)

(подпись, дата)

В.Л. Оленев

(инициалы, фамилия)

Ответственный за ОП ВО 25.03.01(01)

(должность, уч. степень, звание)

(подпись, дата)

Н.И. Ускова

(инициалы, фамилия)

Заместитель директора института №1 по методической работе

ст. преп.

(должность, уч. степень, звание)

(подпись, дата)

В.Е. Таратун

(инициалы, фамилия)

## Аннотация

Дисциплина «Алгоритмизация и программирование» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/ специальности 25.03.01 «Техническая эксплуатация летательных аппаратов и двигателей» направленности «Эксплуатация и испытания авиационной и космической техники». Дисциплина реализуется кафедрой «№14».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

УК-1 «Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач»

Содержание дисциплины охватывает круг вопросов, связанных с разработкой крупных и эффективных системных и сетевых программных продуктов на языке программирования высокого уровня с помощью современных методов и средств разработки и отладки программного обеспечения.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, самостоятельная работа обучающегося.

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

Общая трудоемкость освоения дисциплины составляет 4 зачетных единицы, 144 часа.

Язык обучения по дисциплине «русский»

## 1. Перечень планируемых результатов обучения по дисциплине

### 1.1. Цели преподавания дисциплины

Целью преподавания дисциплины является:

- получение студентами теоретических и практических знаний для формирования навыков разработки эффективных системных и сетевых программных продуктов на языке программирования высокого уровня;
- изучение студентами комплекса сведений, технологических приемов и инструментария создания компонент программного обеспечения, удовлетворяющих современным требованиям к программному продукту;
- подготовка студентов к осознанному использованию языка программирования Си и методов системного и сетевого программирования.
- изучение студентами методов структурного программирования как одного из наиболее распространенных и эффективных методов разработки системных и сетевых программных продуктов;
- изучение студентами методов и механизмов системного и сетевого программирования;
- закрепление студентами навыков работы со средствами разработки и отладки программного обеспечения;
- обучение студентов разработке алгоритмов на основе структурного подхода;
- закрепление навыков алгоритмизации и программирования на основе языка программирования Си;
- изучение студентами основных структур данных и типовых методов обработки этих структур.

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
Универсальные компетенции	УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.3.1 знать методики поиска, сбора и обработки информации, в том числе с использованием информационных технологий УК-1.У.3 уметь оценивать информацию на достоверность; сохранять и передавать данные с использованием цифровых средств УК-1.В.1 владеть навыками критического анализа и синтеза информации, в том числе с помощью цифровых инструментов

## 2. Место дисциплины в структуре ОП

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

– "Информатика".

Знания, полученные при изучении материала данной дисциплины, имеют как самостоятельное значение, так и могут использоваться при изучении других дисциплин.

## 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
1	2	3
<b>Общая трудоемкость дисциплины, ЗЕ/ (час)</b>	4/ 144	4/ 144
<b>Из них часов практической подготовки</b>		
<b>Аудиторные занятия, всего час.</b>	51	51
в том числе:		
лекции (Л), (час)	17	17
практические/семинарские занятия (ПЗ), (час)		
лабораторные работы (ЛР), (час)	34	34
курсовой проект (работа) (КП, КР), (час)		
экзамен, (час)	27	27
<b>Самостоятельная работа, всего (час)</b>	66	66
<b>Вид промежуточной аттестации:</b> зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.**) )	Экз.	Экз.

Примечание: \*\* кандидатский экзамен

#### 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)	ЛР (час)	КП (час)	СРС (час)
Семестр 2					
Раздел 1. Создания и отладка программ.	1		2		4
Раздел 2. Типы, операторы и выражения языка СИ.	1		2		4
Раздел 3. Управление.	1		2		4
Раздел 4. Функции и структура программы.	1		2		4
Раздел 5. Указатели и массивы.	1		2		4
Раздел 6. Структуры.	2		4		4
Раздел 7. Ввод и вывод.	1		2		4
Раздел 8. Методы и механизмы системного программирования.	1		2		4
Раздел 9. Организация функций с переменным числом параметров. Механизм указателя на функции.	1		2		4
Раздел 10. Структуры данных.	1		2		6
Раздел 11. Организация динамических структур данных (абстрактных типов данных): стек, очередь, дек.	2		4		6

Раздел 12. Разработка крупных и эффективных программ	2		4		6
Раздел 13. Стиль программирования.	2		4		6
Раздел 14. Методы и механизмы разработки сетевых приложений.	1		2		6
Итого в семестре:	17		34		66
Итого:	17		34	0	66

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
<b>1</b>	Раздел 1 – Создания и отладка программ. Тема 1.1. Основные этапы решения задач на ЭВМ; критерии качества программы; жизненный цикл программы; постановка задачи и спецификация программы; способы конструирования и верификации программ. Тема 1.2. Обзор языка программирования СИ.
<b>2</b>	Раздел 2 – Типы, операторы и выражения языка СИ. Тема 2.1. Имена переменных, типы и размеры данных, константы, объявления. Тема 2.2. Арифметические операторы, операторы отношения и логические операторы, преобразования типов, операторы инкремент и декремент, побитовые операторы, операторы и выражения присваивания, условные выражения.
<b>3</b>	Раздел 3 - Управление Тема 3.1. Оператор ветвления if, переключатель switch. Тема 3.2. Циклы с предусловием и постусловием. Тема 3.3. Инструкции break, continue, goto.
<b>4</b>	Раздел 4 - Функции и структура программы. Тема 4.1. Основные понятия функции. Тема 4.2. Внешние переменные. Область видимости. Статические переменные, регистровые переменные. Тема 4.3. Рекурсия. Тема 4.4. Препроцессор языка СИ
<b>5</b>	Раздел 5. - Указатели и массивы. Тема 5.1. Указатели и адреса. Тема 5.2. Адресная арифметика.
<b>6</b>	Раздел 6. - Структуры. Тема 6.1. Основные сведения о структурах. Объединения.
<b>7</b>	Раздел 7 – Ввод и вывод. Тема 7.1. Стандартный ввод-вывод и форматный ввод-вывод. Тема 7.2. Доступ к файлам.
<b>8</b>	Раздел 8 – Методы и механизмы системного программирования

	<ol style="list-style-type: none"> <li>1. Системное и прикладное программирование</li> <li>2. Трансляция: компиляция и интерпретация</li> <li>3. Язык C: история и назначение</li> <li>4. Стандартизация Си</li> <li>5. Оператор sizeof</li> <li>6. Выравнивание данных</li> <li>7. Типы size_t, void, void*</li> <li>8. Макрос NULL</li> <li>9. Приведение типов</li> <li>10. Организация памяти программ на Си</li> <li>11. Динамическое распределение памяти</li> <li>12. Работа с динамическим массивом</li> </ol> <p>Многомерные массивы</p>
<b>9</b>	<p>Раздел 9 - Организация функций с переменным числом параметров. Механизм указателя на функции.</p> <ol style="list-style-type: none"> <li>1. ASCII-коды</li> <li>2. Стандартная библиотека языка Си (libc)</li> <li>3. Функции с переменным числом параметров</li> </ol> <p>Указатели на функции</p>
<b>10</b>	<p>Раздел 10 - Структуры данных.</p> <ol style="list-style-type: none"> <li>1. Структуры: <ol style="list-style-type: none"> <li>a. Объявление</li> <li>b. Инициализация</li> <li>c. Операции</li> <li>d. Структуры и функции</li> <li>e. Указатели на структуру</li> <li>f. Массивы структур</li> <li>g. Вложенные структуры</li> </ol> </li> <li>2. typedef</li> <li>3. Препроцессор <ol style="list-style-type: none"> <li>a. Директива #include</li> <li>b. Директива #define <ol style="list-style-type: none"> <li>i. object-like macro</li> <li>ii. function-like macro</li> </ol> </li> <li>c. Условная компиляция</li> <li>d. Определяемые препроцессором константы</li> </ol> </li> </ol>
<b>11</b>	<p>Раздел 11 - Организация динамических структур данных (абстрактных типов данных): стек, очередь, дек.</p> <ol style="list-style-type: none"> <li>1. Однонаправленные списки</li> <li>2. Двухнаправленные списки</li> <li>3. Стеки</li> <li>4. Очереди</li> </ol> <p>Деки</p>
<b>12</b>	<p>Раздел 12 - Разработка крупных и эффективных программ</p> <ol style="list-style-type: none"> <li>1. Аргументы командной строки</li> <li>2. Квалификаторы</li> <li>3. Классы памяти</li> </ol> <p>Сборка проекта из нескольких файлов</p>
<b>13</b>	<p>Раздел 13 - Стиль программирования</p> <ol style="list-style-type: none"> <li>1. Правила наименования</li> <li>2. Табуляция и отступы</li> </ol> <p>Комментарии</p>

<b>14</b>	<p>Раздел 14 - Методы и механизмы разработки сетевых приложений</p> <ol style="list-style-type: none"> <li>1. Ядро сетевого приложения</li> <li>2. Архитектура клиент-сервер</li> <li>3. Средства межпроцессного и межсетевого взаимодействия</li> <li>4. Сокеты: <ol style="list-style-type: none"> <li>a. Использование</li> <li>b. Сокеты и сетевые протоколы</li> <li>c. Типы</li> <li>d. Механизм работы</li> <li>e. Дескриптор</li> <li>f. Атрибуты</li> <li>g. Создание</li> <li>h. Домен</li> <li>i. Адрес</li> <li>j. Связывание</li> <li>k. Закрытие</li> </ol> </li> <li>5. Порядок байт</li> <li>6. Установка соединения</li> <li>7. Передача данных</li> <li>8. Прием данных</li> <li>9. Сокеты в режиме установления соединения</li> </ol> <p>Сокеты в режиме дейтаграмм</p>
-----------	--

#### 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
Учебным планом не предусмотрено					
Всего					

#### 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	Критерии качества программы	1		1
2	Способы конструирования и верификации программ	1		2
3	Инструкции break, continue, goto.	2		3
4	Статические переменные, регистровые переменные.	2		4
5	Обработка символьной информации с	2		5

	использованием указателей			
6	Рекурсия.Адресная арифметика	2		6
7	Простейшие базы данных (работа со структурами)	2		6
8	Работа с файлами	2		7
9	Битовые поля.	2		7
10	Динамическое распределение памяти.	2		8
11	Многомерные массивы	2		8
12	Функции с переменным числом параметров.	2		9
13	Указатели на функции.	2		9
14	Динамические структуры данных	2		10
15	Сборка проекта из нескольких файлов.	2		11
16	Аргументы командной строки.	2		12
17	Архитектура клиент-сервер	2		14
19	Средства межпроцессного и межсетевого взаимодействия	2		14
	Всего	34		

4.5. Курсовое проектирование/ выполнение курсовой работы Учебным планом не предусмотрено

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
1	2	3
Изучение теоретического материала дисциплины (ТО)	30	30
Курсовое проектирование (КП, КР)		
Расчетно-графические задания (РГЗ)		
Выполнение реферата (Р)		
Подготовка к текущему контролю успеваемости (ТКУ)	30	30
Домашнее задание (ДЗ)		
Контрольные работы заочников (КРЗ)		
Подготовка к промежуточной аттестации (ПА)	6	6
Всего:	66	66

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)
004.432 К 36	Керниган Б., Ритчи Д. Язык программирования Си. – Пер. с англ., 2-е изд., перераб. и доп.. - М. и др.: Вильямс, 2006. - 304 с. <a href="http://cpp.com.ru/">http://cpp.com.ru/</a>	1
004.043 С28	Седжвик, Р. Фундаментальные алгоритмы на С Анализ структуры данных. Сортировка. Поиск. Алгоритмы на графах: Пер. с англ/ Р. Седжвик. - 3-е изд. - М. и др.: DiaSoft, 2003. - 1127 с	22
	Хэзфилд Р., Кирби Л. Искусство программирования на С: Фундаментальные алгоритмы, структуры данных и примеры приложений	
	Стив Саммит Язык С в вопросах и ответах	
	Стивен Прата. Язык программирования С. Лекции и упражнения, 5-е издание	
	К. N. King C Programming: A Modern Approach	

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
<a href="https://docs.google.com/document/d/1oKApRCt_ukeSMbqlsC2u-Wzzlfl_2ysbJwMzDKojEs">https://docs.google.com/document/d/1oKApRCt_ukeSMbqlsC2u-Wzzlfl_2ysbJwMzDKojEs</a>	Осмоловский С.В. Методические указания к выполнению лабораторных работ (ЛР 1-4). Учебное пособие в электронном виде (Инф. ресурс кафедры). 2015 год

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
1	Операционная система семейства MS Windows (XP, Vista, 7, 8, 10) или семейства Linux
2	Среда разработки и отладки проектов на языке программирования высокого уровня Си (Visual Studio, Codeblocks, Qt Creator, NetBeans, Eclipse и т.д.), компилятор gcc или аналогичный по функциональности.
3	Пакета MS Office или аналогичные по функциональности

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 таблице12.

Таблица 12 – Состав материально-технической базы

№ п/п	Наименование составной части материально-технической базы	Номер аудитории (при необходимости)
1	Мультимедийная лекционная аудитория	
2	Компьютерный класс	

#### 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

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 5-балльная шкала	Характеристика сформированных компетенций
«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; – 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;

Оценка компетенции 5-балльная шкала	Характеристика сформированных компетенций
	– 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	Тип void. Указатель на void (void *).	
2	Макрос NULL. Тип size t. Оператор sizeof ().	
3	Приведение типов. Приведение при работе с указателями.	
4	Динамическое распределение памяти. Функции выделения динамической памяти (на примере многомерных массивов).	
5	Динамическое распределение памяти. Функции перераспределения и освобождения динамической памяти (на примере многомерных массивов).	
6	Статические и динамические многомерные массивы. Расположения в памяти.	
7	Многомерные массивы. Способы передачи статических многомерных массивов в функции.	
8	Многомерные массивы. Способы передачи динамических многомерных массивов в функции.	
9	Многомерные массивы. Массивы указателей. Указатель на указатель.	
10	Организация оперативной памяти для программ на Си.	
11	ASCII-коды. Таблица ASCII-символов. Управляющие последовательности.	
12	Стандартная библиотека языка Си (libc): работа со строками.	
13	Стандартная библиотека языка Си (libc): работа с областями памяти, преобразование типов.	
14	Функции с переменным числом параметров: назначение, применение, механизмы доступа к аргументам функции, способы определения размера переменного списка	

	параметров.	
15	Указатели на функции: определение, объявление, применение. Таблица функций.	
16	Препроцессор. Основные препроцессорные директивы и их назначение. Директива #include.	
17	Препроцессор. Макросы (директива #define с параметрами и без). Условная компиляция.	
18	Структуры. Структуры и функции. Указатели на структуры.	
19	Структуры. Массивы структур. Вложенные структуры.	
20	Оператор typedef.	
21	Динамические структуры данных. Однонаправленные списки: определение, особенности, основные операции.	
22	Динамические структуры данных. Двухнаправленные списки: определение, особенности, основные операции.	
23	Динамические структуры данных. Стек: определение, особенности, основные операции.	
24	Динамические структуры данных. Очередь: определение, особенности, основные операции.	
25	Динамические структуры данных. Дек: определение, особенности, основные операции.	
26	Аргументы командной строки. Аргумент argc: описание, особенности. Примеры.	
27	Аргументы командной строки. Аргумент argv: описание, особенности. Примеры.	
28	Квалификатор const: описание, применение, примеры использования.	
29	Квалификатор volatile: описание, применение, примеры использования.	
30	Квалификатор функции inline: описание, применение, примеры использования.	
31	Классы памяти: перечисление, применение. Время жизни и область видимости: определение. Виды классов памяти.	
32	Классы памяти. Автоматические переменные: описание, особенности, применение.	
33	Классы памяти. Регистровые переменные: особенности, ограничения.	
34	Классы памяти. Глобальные и локальные переменные с квалификатором static: особенности, примеры.	
35	Классы памяти. Классификатор extern: особенности, применение с переменными и функциями. Примеры.	
36	Сборка программы из нескольких файлов: этапы сборки, их описание и схематический рисунок. Модульная структура. Заголовочные файлы: применение, ограничения.	
37	Определить массив указателей на функцию, имеющий два параметра типа int и возвращающую результат типа double.	
38	Что будет выведено на экран в 32-разрядной операционной системе при использовании компилятора Си:	

	char c = 'a'; printf ("%d\t", sizeof (char)); printf ("%d\t", sizeof (char *)); printf ("%d\t", sizeof (c)); printf ("%d\t", sizeof ('a')); printf ("%d\n", sizeof ("Hello, world!"));	
39	Определить указатель на функцию без контроля параметров вызова, возвращающую целое значение.	
40	Выделить память для двумерного динамического массива со значениями с плавающей точкой, используя указатель на указатель.	
41	Пусть существует 4 функции, которые принимают два числа типа float. Прототипы функций представлены ниже: 1. float sta(float a, float b); 2. float div(float a, float b); 3. float add(float a, float b); 4. float sub(float a, float b); Требуется объявить таблицу функций, элементы которой будут указатели на вышеприведенные функции.	
42	Имитировать работу с двумерным символьным массивом через динамически выделенный одномерный массив.	
43	В 32-разрядной операционной системе при использовании компилятора Си определить минимально возможное значение, которое вернет оператор sizeof, примененный к имени типа структуры: struct { double h[5]; int b; } str; int sz; sz = sizeof (str);	
44	С помощью typedef определить новое имя типа указателя на функцию, принимающую два символьных аргумента и возвращающую целое значение. Используя данное имя, объявить указатель на соответствующую функцию.	
45	С помощью typedef определить новое имя типа указателя на функцию, принимающую два целочисленных аргумента и возвращающую символьное значение. Используя данное имя, объявить массив указателей на соответствующие функции из 5 элементов.	
46	С помощью typedef определить новое имя типа структуры с двумя целочисленными полями и символьным полем из 60 элементов. Используя данное имя, создать переменную данного типа.	

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
	Учебным планом не предусмотрено

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

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.

## 11. Методические указания для обучающихся по освоению дисциплины

11.1. 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, приходить 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:

- Изложение лекционного материала;
- Освоение теоретического материала по вопросам, представленным в таблице 19;
- Список вопросов по теме для самостоятельной работы студента.

11.2. Методические указания для обучающихся по выполнению 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 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

- Каждая ЛР выполняется по индивидуальному заданию, выданному студенту преподавателем;
- В задании должно быть четко сформулирована задача, выполняемая в ЛР;
- ЛР должна выполняться на основе полученных теоретических знаний;
- Выполнение ЛР должно осуществляться на основе методических указаний, предоставляемых преподавателем;
- ЛР должна выполняться в специализированном компьютерном классе и может быть доработана студентом в домашних условиях, если позволяет ПО;
- Итогом выполненной ЛР является отчет и демонстрация результатов работы преподавателю в электронном виде.

### Структура и форма отчета о лабораторной работе

- Постановка задачи;
- Содержание этапов выполнения;
- Листинг программы на ЯВУ Си.
- Контрольные тесты. Подраздел содержит наборы исходных данных и полученные в ходе выполнения программы результаты.
- Обоснование полученного результата (вывод);

Т.к. итогом выполнения ЛР является не отчет, а демонстрация результатов работы в электронном виде, то студент должен продемонстрировать преподавателю, как получены результаты работы.



Требования к оформлению отчета о лабораторной работе

- Лабораторная работа (ЛР) предоставляется в печатном/или электронном виде;
- ЛР должна иметь титульный лист (ГОСТ 7.32-2001 издания 2008 года) с названием и подписью студента, который ее сделал и оформил;
- Студент должен защитить ЛР. Отметка о защите должна находиться на титульном листе вместе с подписью преподавателя.

## 11.3. 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.

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.

## 11.4. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

## 11.5. 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».

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