

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
Ивангородский гуманитарно-технический институт (филиал)
федерального государственного автономного образовательного учреждения высшего
образования
"Санкт-Петербургский государственный университет аэрокосмического
приборостроения"

Кафедра № 2

УТВЕРЖДАЮ
Руководитель образовательной программы

старший преподаватель

(должность, уч. степень, звание)

А.А. Сорокин

(инициалы, фамилия)



(подпись)

«25» февраля 2026 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Основы программирования»
(Наименование дисциплины)

Код направления подготовки/ специальности	09.03.01
Наименование направления подготовки/ специальности	Информатика и вычислительная техника
Наименование направленности/ специализации	Программное обеспечение средств вычислительной техники и автоматизированных систем
Форма обучения	очно-заочная
Год приема	2026

Ивангород – 2026

Лист согласования рабочей программы дисциплины

Программу составил (а)

доцент, к.т.н., доцент
(должность, уч. степень, звание)

10.02.2026
(подпись, дата)

Л.Н. Бариков
(инициалы, фамилия)

Программа одобрена на заседании кафедры № 2

«10» февраля 2026 г, протокол № 6

И.о. зав. кафедрой № 2

д.ф.-м.н.
(уч. степень, звание)

10.02.2026
(подпись, дата)

Ю.В. Рождественский
(инициалы, фамилия)

Заместитель директора ИФ ГУАП по методической работе

(должность, уч. степень, звание)

10.02.2026
(подпись, дата)

Н.В. Шустер
(инициалы, фамилия)

Аннотация

Дисциплина «Основы программирования» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/специальности 09.03.01 «Информатика и вычислительная техника» направленности/специализации «Программное обеспечение средств вычислительной техники и автоматизированных систем». Дисциплина реализуется кафедрой «№2».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ОПК-1 «Способен применять естественнонаучные и общинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности»

ОПК-2 «Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности»

ОПК-8 «Способен разрабатывать алгоритмы и программы, пригодные для практического применения»

Содержание дисциплины охватывает круг вопросов, связанных с современным состоянием и тенденциями развития языков программирования и средств для разработки программ различного уровня сложности (принципы разработки, написания и отладки программ разной степени сложности, основы системного программирования, основы объектно-ориентированного подхода к программированию, принципы действия, основные функции и интерфейс современных инструментальных средств).

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

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме экзамена (2 семестр), экзамена (3 семестр).

Общая трудоемкость освоения дисциплины составляет 11 зачетных единиц, 396 часов.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Формирование у студентов представления о современном состоянии языков программирования и средствах для разработки программ различного уровня сложности, а также развитие практических навыков по разработке программ на языках программирования высокого уровня с использованием сред визуального объектно-ориентированного программирования.

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-1 Способен применять естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	ОПК-1.3.1 знать основы математики, физики, вычислительной техники и программирования ОПК-1.У.1 уметь решать стандартные профессиональные задачи с применением естественнонаучных и обще-технических знаний, методов математического анализа и моделирования
Общепрофессиональные компетенции	ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.3.1 знать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности
Общепрофессиональные компетенции	ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-8.3.1 знать алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения ОПК-8.У.1 уметь составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы,

		интегрировать программные модули ОПК-8.В.1 владеть языком программирования; навыками отладки и тестирования работоспособности программы
--	--	---

2. Место дисциплины в структуре ОП

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

- Информатика;
- Математика. Аналитическая геометрия и линейная алгебра.

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

- Базы данных;
- Защита информации;
- Операционные системы;
- Сети ЭВМ и телекоммуникации;
- Технология программирования;
- Электроника;
- Электротехника.

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)	11/ 396	7/ 252	4/ 144
Из них часов практической подготовки			
Аудиторные занятия, всего час.	102	51	51
в том числе:			
лекции (Л), (час)	51	34	17
практические/семинарские занятия (ПЗ), (час)			
лабораторные работы (ЛР), (час)	34	17	17
курсовой проект (работа) (КП, КР), (час)	17		17
экзамен, (час)	99	54	45
Самостоятельная работа, всего (час)	195	147	48
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.)	Экз., Экз., Курс. Раб.	Экз.,	Экз., Курс. Раб.

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/КР (час)	СР (час)
Семестр 2					
Раздел 1. Современные методы разработки программных средств Тема 1.1. Современные методы разработки программных средств Тема 1.2. Этапы разработки программных средств	6		2		37
Раздел 2. Основы языка программирования C++ Тема 2.1. Современные языки программирования Тема 2.2. Базовые средства языка C++ Тема 2.3. Понятие объекта в C++	6		0		25
Раздел 3. Структурное программирование Тема 3.1. Базовые конструкции структурного программирования Тема 3.2. Примеры разработки программ Тема 3.3. Статические массивы	6		15		60
Раздел 4. Процедурное и модульное программирование Тема 4.1. Концепции процедурного программирования Тема 4.2. Понятие рекурсии Тема 4.3. Концепции модульного программирования	16		0		25
Итого в семестре:	34	0	17		147
Семестр 3					
Раздел 5. Динамические структуры данных Тема 5.1. Динамические структуры данных Тема 5.2. Деревья	9	0	15		19
Раздел 6. Концепции объектно-ориентированного программирования Тема 6.1. Классы Тема 6.2. Перегрузка операций Тема 6.3. Наследование Тема 6.4. Потокотковые классы Тема 6.5. Строки в стиле C++ Тема 6.6. Основы доказательства правильности программ	9	0	2		10
Выполнение курсовой работы				17	19
Итого в семестре:	17		34	17	48
Итого	51	0	34	17	195

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
1	Современные методы разработки программных средств Тема 1.1. Современные методы разработки программных средств Развитие методов программирования. Структурное, процедурное, модульное и объектно-ориентированное программирование в контексте C++. Технологии нисходящего и восходящего проектирования. Сущность структурного программирования: декомпозиция задач, стандартные управляющие конструкции. Достоинства и недостатки структурного подхода в C++. Тема 1.2. Этапы разработки программных средств Этапы разработки ПО: постановка задачи, проектирование, кодирование, тестирование, документирование. Проектирование структур данных: входные/выходные данные, интерфейсы. Использование нисходящего проектирования при создании программ на C++. Этап структурного программирования. Этап тестирования и отладки (gdb, sanitizers, unit-тесты).
2	Основы языка программирования C++ Тема 2.1. Современные языки программирования Понятие языка программирования. Этапы развития языков: от C к C++98, C++11, C++14, C++17, C++20, C++23. Современные тенденции: шаблоны, метапрограммирование, RAII, безопасная работа с памятью. Сравнение языков по модели данных и способам реализации алгоритмов. Синтаксис и семантика. Способы описания синтаксиса: BNF, диаграммы. Тема 2.2. Базовые средства языка C++ Структура программы на C++. Алфавит, лексемы, выражения, комментарии. Типы данных: целочисленные, вещественные, логические, символьные. Область видимости и время жизни объектов. Ключевые слова. Константы, литералы, спецификаторы хранения (static, extern, register). Операторы языка: арифметические, логические, поразрядные, присваивания. Структура программы: функции, заголовочные файлы, пространство имён. Компиляция и линковка. Тема 2.3. Понятие объекта в C++ Объекты и их классификация. Арифметические операции над объектами. Преобразования типов: неявные, явные

	(static_cast, reinterpret_cast, const_cast, dynamic_cast). Операции с указателями. Ссылки. Операции разыменования, взятия адреса. Порядок вычисления выражений. Операции инкремента/декремента. Условные выражения. Операция запятая. Размер объектов (sizeof). Понятие выражения и его тип.
3	Структурное программирование Тема 3.1. Базовые конструкции структурного программирования Линейные алгоритмы: выражения, блоки, пустой оператор. Разветвления: if, switch, break, continue, return. Циклы: while, do...while, for. Итерационные и вложенные циклы. Рекуррентные вычисления. Тема 3.2. Примеры разработки программ Линейные программы. Поразрядные операции над целыми числами. Разветвляющиеся программы: поиск максимумов/минимумов. Геометрические задачи: попадание точки в область. Циклические программы: – фиксированное число повторений, – вычисления с заданной точностью, – таблицы значений функций. Тема 3.3. Статические массивы Статическое распределение памяти. Структурированные типы данных: одномерные и многомерные массивы. Решение типовых задач на массивы: поиск значений максимального отрицательного и минимального положительного элементов одномерного вещественного массива, перестановка элементов одномерного массива. Перестановка столбцов и строк матриц по заданным правилам.
4	Процедурное и модульное программирование Тема 4.1. Концепции процедурного программирования Функции. Основные понятия. Принципы использования функций в программах. Объявление и определение функций. Список параметров и тип функции. Вызов функций на исполнение. Формальные и фактические параметры. Области действия описаний функций. Локальность и глобальность. Механизм передачи параметров. Глобальные переменные. Передача массивов в качестве параметров. Примеры. Реализация алгоритмов сортировки структур данных и поиска в этих структурах. Тема 4.2. Понятие рекурсии Передача имен функций в качестве параметров. Понятие рекурсии. Рекурсивные определения и алгоритмы. Программирование рекурсивных алгоритмов: рекурсивные функции. Механизм рекурсивных вызовов. Перегрузка

	функций. Шаблоны функций. Тема 4.3. Концепции модульного программирования Модули в C++: заголовочные файлы, единицы трансляции. Инкапсуляция через namespace. Сборка многомодульных программ. Система сборки (CMake). Модули C++20.
5	Динамические структуры данных Тема 5.1. Динамические структуры данных Указатели и ссылки. Динамическое выделение памяти (new, delete). Умные указатели (unique_ptr, shared_ptr, weak_ptr). Динамические массивы (std::vector). Списки: односвязные, двусвязные, циклические. Стек (std::stack). Очередь (std::queue). Дек (std::deque). Тема 5.2. Деревья Бинарные деревья. Рекурсивная структура дерева. Обходы: инфиксный, префиксный, постфиксный. Построение дерева. Деревья поиска. Балансировка (обзорно).
6	Концепции объектно-ориентированного программирования Тема 6.1. Классы Инкапсуляция, наследование, полиморфизм. Классы: поля, методы, доступ (public, private, protected). Объекты и их время жизни. Конструкторы: виды, список инициализации. Статические поля и методы. Дружественные функции и классы. Деструкторы. Правило трёх/пяти. Тема 6.2. Перегрузка операций Перегрузка унарных и бинарных операций. Перегрузка присваивания. Перегрузка приведения типа. Перегрузка вызова функции (operator()). Перегрузка индексирования (operator[]). Перегрузка ввода/вывода (<<, >>). Тема 6.3. Наследование Простое и множественное наследование. Виртуальные методы. Позднее связывание. Абстрактные классы. Шаблоны классов (class templates). Использование шаблонов. Тема 6.4. Потокосовые классы Стандартные потоки ввода/вывода (iostream). Файловые потоки (fstream). Текстовые и бинарные файлы. Перегрузка операторов для потоков. Тема 6.5. Строки в стиле C++ Типы строк: char*, std::string, std::wstring. Операции со строками. Функции преобразования. Поиск подстрок. Сравнение строк. Получение характеристик строк. Тема 6.6. Основы доказательства правильности программ Теорема структуры и структурное программирование. Анализ программ. Утверждения о программах. Корректность программ. Правила вывода для основных структур

	программирования. Инвариантные утверждения. Утверждения о массивах. Способы верификации программ. Жизненный цикл программы.
--	---

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	Следование	2		1
2	Поиск экстремума	2		3
3	Операторы ветвления	2		3
4	Операторы цикла: заданное число повторений	2		3
5	Операторы цикла: вычисления с заданной точностью	2		3
6	Суммирование рядов	2		3
7	Перегрузка функций	2		3
8	Статические одномерные и двумерные массивы	3		3
Семестр 3				
9	Динамические массивы	2		5
10	Методы сортировки	2		5
11	Передача имён функций в качестве параметра	2		5
12	Строки и текстовые файлы	2		5
13	Структуры и бинарные файлы	2		5
14	Линейные списки	2		5
15	Стек, дек, очередь	3		5
16	Классы	2		6
Всего		34		

4.5. Выполнение курсового проекта/ курсовой работы

Цель курсовой работы: Цель курсового проекта: систематизация, закрепление и расширение теоретических знаний и практических навыков, полученных в рамках учебного плана направления 09.03.01 и применение этих знаний и навыков при решении конкретных научно-исследовательских, инженерно-технических, организационных и производственных задач.

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)	126	115	21
Курсовое проектирование (КП, КР)	19		19
Расчетно-графические задания (РГЗ)			
Выполнение реферата (Р)			
Подготовка к текущему контролю успеваемости (ТКУ)	20	16	4
Домашнее задание (ДЗ)			
Контрольные работы заочников (КРЗ)			
Подготовка к промежуточной аттестации (ПА)	20	16	4
Всего:	195	147	48

5. Перечень учебно-методического обеспечения

для самостоятельной работы обучающихся по дисциплине (модулю)

Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. разделов 6-11.

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)
https://znanium.ru/catalog/product/2162065 Режим доступа: для авторизованных пользователей.	Шитов, В. Н. Программирование на примере C++ : учебное пособие / В.Н. Шитов. — Москва : ИНФРА-М, 2025. — 412 с.	-
https://znanium.ru/catalog/	Демяненко, Я. М. C++ как второй язык в	-

product/2239841 Режим доступа: для авторизованных пользователей.	обучении приемам и технологиям программирования : учебное пособие / Я. М. Демяненко, М. И. Чердынцева. – 2-е изд., испр. и доп. - Южный федеральный университет. – Ростов-на-Дону ; Таганрог: Издательство Южного федерального университета, 2025. - 423 с.	
https://e.lanbook.com/book/515057 Режим доступа: для авторизованных пользователей.	Карпович, Е. Е. Основы алгоритмизации и программирования : учебное пособие для вузов / Е. Е. Карпович. — Санкт-Петербург : Лань, 2026. — 144 с.	-
https://e.lanbook.com/book/506533 Режим доступа: для авторизованных пользователей.	Соколова, А. Н. Основы программирования на языке C++ : учебно-методическое пособие / А. Н. Соколова, А. В. Торбеева, Н. В. Шалагинова. — Киров : ВятГУ, 2024. — 220 с.	-

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
https://www.intuit.ru/	Национальный Открытый Университет "ИНТУИТ"
http://lib.guap.ru/	Библиотека ГУАП
http://pro.guap.ru/	Личный кабинет ГУАП
http://lms.guap.ru/	Система дистанционного образования ГУАП
https://znanium.com/	Электронно-библиотечная система Znanium
https://urait.ru/	Образовательная платформа Юрайт

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	Embarcadero RAD Studio XE7 Professional
2	Microsoft Visual Studio Community
3	Microsoft Visual Code

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	Фонд аудиторий ИФ ГУАП для проведения лекционных и практических (семинарских) занятий	
2	Лаборатория программирования и баз данных	207
3	Кабинет информационных технологий и программных систем	212
4	Помещения для организации самостоятельной работы	111

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; Тесты.
Выполнение курсовой работы	Экспертная оценка на основе требований к содержанию курсовой работы по дисциплине.

Примечание: *экзаменационные билеты формируются на основе вопросов и задач таблицы 15.

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 связывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий. – правильно выполнил от 90% до 100% тестовых заданий.
«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. – правильно выполнил от 70% до 89% тестовых заданий.
«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. – правильно выполнил от 51% до 69% тестовых заданий.
«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. – правильно выполнил менее 51% тестовых заданий.

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	Что включает в себя понятие «основы программирования»	ОПК-1.3.1
2	Как связаны математика, физика и программирование	ОПК-1.3.1
3	Что такое вычислительная техника и как она используется в программировании	ОПК-1.3.1
4	Какие этапы включает решение инженерной задачи с использованием программирования	ОПК-1.У.1
5	Как применяется математическое моделирование при разработке программ	ОПК-1.У.1

6	Какие современные информационные технологии используются в программировании	ОПК-2.3.1
7	Какие отечественные программные средства применяются в разработке ПО	ОПК-2.3.1
8	Как выбрать подходящую ИТ-технологии для решения задачи	ОПК-2.У.1
9	Что такое алгоритмический язык программирования	ОПК-8.3.1
10	Какие этапы развития прошёл язык С++	ОПК-8.3.1
11	Что такое структурное программирование	ОПК-8.3.1
12	Что такое процедурное программирование	ОПК-8.3.1
13	Что такое модульное программирование	ОПК-8.3.1
14	Что такое объектно-ориентированное программирование	ОПК-8.3.1
15	Что такое нисходящее и восходящее проектирование	ОПК-1.3.1
16	Какие этапы включает разработка программного обеспечения	ОПК-1.3.1
17	Что такое структура программы на С++	ОПК-8.3.1
18	Какие типы данных поддерживает С++	ОПК-8.3.1
19	Что такое область видимости и время жизни объектов	ОПК-8.3.1
20	Какие существуют операторы языка С++	ОПК-8.3.1
21	Что такое объект в С++	ОПК-8.3.1
22	Что такое преобразование типов	ОПК-8.3.1
23	Что такое указатель и ссылка	ОПК-8.3.1
24	Что такое выражение и его тип	ОПК-8.3.1
25	Какие конструкции относятся к структурному программированию	ОПК-8.3.1
26	Что такое линейный алгоритм	ОПК-8.3.1
27	Что такое разветвляющийся алгоритм	ОПК-8.3.1
28	Что такое цикл и какие виды циклов существуют	ОПК-8.3.1
29	Что такое статический массив	ОПК-8.3.1
30	Какие задачи решаются с помощью массивов	ОПК-1.У.1
31	Что такое функция в С++	ОПК-8.3.1
32	Что такое параметры функции и их виды	ОПК-8.3.1
33	Что такое рекурсия	ОПК-8.3.1
34	Что такое перегрузка функций	ОПК-8.3.1
35	Что такое шаблоны функций	ОПК-8.3.1
36	Что такое модуль в С++	ОПК-8.3.1
37	Что такое единица трансляции	ОПК-8.3.1
38	Что такое система сборки СMake	ОПК-2.3.1
39	Что такое динамическое выделение памяти	ОПК-8.3.1
40	Что такое умные указатели	ОПК-8.3.1
41	Что такое динамические структуры данных	ОПК-8.3.1
42	Что такое бинарное дерево	ОПК-8.3.1
43	Что такое обходы дерева	ОПК-8.3.1
44	Что такое инкапсуляция	ОПК-8.3.1
45	Что такое наследование	ОПК-8.3.1
46	Что такое полиморфизм	ОПК-8.3.1
47	Что такое конструктор и деструктор	ОПК-8.3.1
48	Что такое перегрузка операторов	ОПК-8.3.1
49	Что такое виртуальные методы	ОПК-8.3.1
50	Что такое абстрактный класс	ОПК-8.3.1
51	Что такое шаблоны классов	ОПК-8.3.1

52	Что такое поток ввода/вывода	ОПК-8.3.1
53	Чем отличаются текстовые и бинарные файлы	ОПК-8.3.1
54	Что такое std::string и std::wstring	ОПК-8.3.1
55	Какие операции выполняются над строками	ОПК-8.3.1
56	Что такое корректность программы	ОПК-1.3.1
57	Что такое инвариант цикла	ОПК-1.3.1
58	Какие существуют методы верификации программ	ОПК-1.3.1
59	Как составлять алгоритмы и писать программы	ОПК-8.У.1
60	Как тестировать и отлаживать программы	ОПК-8.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 выполнения курсового проекта/ курсовой работы представлены в таблице 17.

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

№ п/п	Примерный перечень тем для выполнения курсового проекта/ курсовой работы
1	Транспортная задача.
2	Задача о назначениях.
3	Задача об оптимальном ассортименте.
4	Задача о диете.
5	Задача о максимальном потоке.
6	Задача одного исполнителя на минимакс.
7	Задача одного исполнителя на минисумму.
8	Задача о нескольких исполнителях.
9	Задача о дорожной сети.
10	Задача о телефонной связи.
11	Задача строительной трассировки.
12	Задача электронной трассировки.
13	Задача размещения.
14	Поиск в глубину графа.
15	Поиск в ширину графа.
16	Кратчайшие пути в графе.
17	Эйлеровы пути.
18	Гамильтоновы циклы.
19	База данных телефонной сети.
20	База данных библиотеки.
21	База данных магазина грамзаписей.
22	Игра «Морской бой».
23	Игра «Sudoku».
24	Игра «Змейка».
25	Игра «Поле чудес».

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								
1	Задание закрытого типа с выбором одного правильного ответа Инструкция: Прочитайте текст и выберите один правильный ответ Какой принцип лежит в основе структурного программирования? 1. Рекурсивное наследование 2. Декомпозиция задач и использование базовых управляющих конструкций 3. Динамическая диспетчеризация 4. Множественное наследование Ключ с правильным ответом: 2	ОПК-1.3.1								
2	Задание закрытого типа с выбором нескольких правильных ответов Инструкция: Прочитайте текст и выберите все правильные ответы Какие элементы относятся к базовым конструкциям структурного программирования? 1. Последовательность 2. Разветвление 3. Цикл 4. Шаблон класса Ключ с правильным ответом: 1, 2, 3	ОПК-1.3.1								
3	Задание закрытого типа на установление соответствия Инструкция: Прочитайте текст и установите соответствие Сопоставьте этап разработки ПО и его содержание. <table><tr><th>Этап</th><th>Содержание</th></tr><tr><td>А. Постановка задачи</td><td>1. Определение входных и выходных данных</td></tr><tr><td>Б. Кодирование</td><td>2. Реализация алгоритмов на языке C++</td></tr><tr><td>В. Тестирование</td><td>3. Поиск и исправление ошибок</td></tr></table> Ключ с правильным ответом: А1, Б2, В3	Этап	Содержание	А. Постановка задачи	1. Определение входных и выходных данных	Б. Кодирование	2. Реализация алгоритмов на языке C++	В. Тестирование	3. Поиск и исправление ошибок	ОПК-1.3.1
Этап	Содержание									
А. Постановка задачи	1. Определение входных и выходных данных									
Б. Кодирование	2. Реализация алгоритмов на языке C++									
В. Тестирование	3. Поиск и исправление ошибок									
4	Задание закрытого типа на установление правильной последовательности Инструкция: Прочитайте текст и установите последовательность. Этапы разработки программного обеспечения: 1. Постановка задачи 2. Проектирование 3. Кодирование 4. Тестирование 5. Документирование Запишите последовательность цифр слева направо. Ключ с правильным ответом: 1, 2, 3, 4, 5	ОПК-1.3.1								
5	Задание открытого типа Инструкция: Прочитайте текст и запишите ответ. Как называется механизм, позволяющий программе автоматически освобождать ресурсы при выходе объекта из области видимости в C++? Ключ с правильным ответом: RAII	ОПК-1.3.1								
6	Задание открытого типа с развернутым обоснованным ответом Инструкция: Прочитайте текст и запишите развернутый обоснованный ответ Объясните, почему структурное программирование облегчает анализ корректности программ.	ОПК-1.3.1								

	Ключ с правильным ответом (эталонный ответ): Структурное программирование использует ограниченный набор управляющих конструкций и чёткую декомпозицию задач, что делает поток управления предсказуемым. Это упрощает доказательство корректности, анализ инвариантов и проверку выполнения условий, обеспечивая формальную проверяемость программ.									
7	Задание закрытого типа с выбором одного правильного ответа Инструкция: Прочитайте текст и выберите один правильный ответ Какой цикл используется, когда количество повторений заранее неизвестно? 1. for 2. while 3. switch 4. goto Ключ с правильным ответом: 2	ОПК-1.У.1								
8	Задание закрытого типа с выбором нескольких правильных ответов Инструкция: Прочитайте текст и выберите все правильные ответы Какие задачи относятся к типовым задачам структурного программирования? 1. Поиск максимального элемента массива 2. Проверка попадания точки в область 3. Построение бинарного дерева поиска 4. Вычисление таблицы значений функции Ключ с правильным ответом: 1, 2, 4	ОПК-1.У.1								
9	Задание закрытого типа на установление соответствия Инструкция: Прочитайте текст и установите соответствие Сопоставьте тип программы и пример задачи. <table><tr><td>Тип программы</td><td>Пример</td></tr><tr><td>А. Линейная</td><td>1. Вычисление выражения</td></tr><tr><td>Б. Разветвляющаяся</td><td>2. Определение максимума из трёх чисел</td></tr><tr><td>В. Циклическая</td><td>3. Вычисление суммы ряда</td></tr></table> Ключ с правильным ответом: А1, Б2, В3	Тип программы	Пример	А. Линейная	1. Вычисление выражения	Б. Разветвляющаяся	2. Определение максимума из трёх чисел	В. Циклическая	3. Вычисление суммы ряда	ОПК-1.У.1
Тип программы	Пример									
А. Линейная	1. Вычисление выражения									
Б. Разветвляющаяся	2. Определение максимума из трёх чисел									
В. Циклическая	3. Вычисление суммы ряда									
10	Задание закрытого типа на установление правильной последовательности Инструкция: Прочитайте текст и установите последовательность. Этапы решения задачи программирования: 1. Анализ условия 2. Разработка алгоритма 3. Кодирование 4. Тестирование 5. Оценка корректности Запишите последовательность цифр слева направо. Ключ с правильным ответом: 1, 2, 3, 4, 5	ОПК-1.У.1								
11	Задание открытого типа Инструкция: Прочитайте текст и запишите ответ. Как называется оператор, позволяющий выбирать один из нескольких вариантов в зависимости от значения выражения? Ключ с правильным ответом: switch	ОПК-1.У.1								
12	Задание открытого типа с развернутым обоснованным ответом Инструкция: Прочитайте текст и запишите развернутый обоснованный ответ Объясните, почему использование циклов позволяет решать задачи	ОПК-1.У.1								

	вычисления с заданной точностью. Ключ с правильным ответом (эталонный ответ): Циклы позволяют выполнять повторяющиеся вычисления до достижения определённого критерия остановки. Это обеспечивает возможность контролировать точность результата, постепенно уточняя вычисления и прекращая процесс, когда погрешность становится меньше заданного порога.									
13	Задание закрытого типа с выбором одного правильного ответа Инструкция: Прочитайте текст и выберите один правильный ответ Какой инструмент используется для сборки многомодульных проектов в C++? 1. make 2. CMake 3. gdb 4. valgrind Ключ с правильным ответом: 2	ОПК-2.3.1								
14	Задание закрытого типа с выбором нескольких правильных ответов Инструкция: Прочитайте текст и выберите все правильные ответы Какие средства относятся к инструментам отладки программ? 1. gdb 2. sanitizers 3. unit-тесты 4. компоновщик ld Ключ с правильным ответом: 1, 2, 3	ОПК-2.3.1								
15	Задание закрытого типа на установление соответствия Инструкция: Прочитайте текст и установите соответствие Сопоставьте инструмент и его назначение. <table><tr><td>Инструмент</td><td>Назначение</td></tr><tr><td>A. gdb</td><td>1. Пошаговая отладка</td></tr><tr><td>Б. CMake</td><td>2. Генерация файлов сборки</td></tr><tr><td>В. valgrind</td><td>3. Анализ утечек памяти</td></tr></table> Ключ с правильным ответом: A1, B2, B3	Инструмент	Назначение	A. gdb	1. Пошаговая отладка	Б. CMake	2. Генерация файлов сборки	В. valgrind	3. Анализ утечек памяти	ОПК-2.3.1
Инструмент	Назначение									
A. gdb	1. Пошаговая отладка									
Б. CMake	2. Генерация файлов сборки									
В. valgrind	3. Анализ утечек памяти									
16	Задание закрытого типа на установление правильной последовательности Инструкция: Прочитайте текст и установите последовательность. Этапы тестирования программы: 1. Подготовка тестовых данных 2. Запуск программы 3. Анализ результатов 4. Исправление ошибок 5. Повторное тестирование Запишите последовательность цифр слева направо. Ключ с правильным ответом: 1, 2, 3, 4, 5	ОПК-2.3.1								
17	Задание открытого типа Инструкция: Прочитайте текст и запишите ответ. Как называется технология, позволяющая автоматически генерировать файлы сборки для разных платформ? Ключ с правильным ответом: CMake	ОПК-2.3.1								
18	Задание открытого типа с развернутым обоснованным ответом Инструкция: Прочитайте текст и запишите развернутый обоснованный ответ Объясните, почему использование unit-тестов повышает	ОПК-2.3.1								

	надёжность программного обеспечения. Ключ с правильным ответом (эталонный ответ): Unit-тесты проверяют корректность отдельных модулей программы в изоляции. Это позволяет выявлять ошибки на ранних этапах, предотвращать регрессии и обеспечивать стабильность системы при внесении изменений.									
19	Задание закрытого типа с выбором одного правильного ответа Инструкция: Прочитайте текст и выберите один правильный ответ Какой стандарт языка C++ впервые ввёл ключевое слово constexpr? 1. C++98 2. C++11 3. C++14 4. C++17 Ключ с правильным ответом: 2	ОПК-8.3.1								
20	Задание закрытого типа с выбором нескольких правильных ответов Инструкция: Прочитайте текст и выберите все правильные ответы Какие элементы относятся к современным средствам разработки ПО? 1. IDE 2. Система контроля версий 3. Компилятор 4. Текстовый редактор без подсветки Ключ с правильным ответом: 1, 2, 3	ОПК-8.3.1								
21	Задание закрытого типа на установление соответствия Инструкция: Прочитайте текст и установите соответствие Сопоставьте компонент среды разработки и его назначение. <table><tr><td>Компонент</td><td>Назначение</td></tr><tr><td>А. Компилятор</td><td>1. Преобразование исходного кода в машинный</td></tr><tr><td>Б. Отладчик</td><td>2. Пошаговое выполнение программы</td></tr><tr><td>В. Система контроля версий</td><td>3. Управление изменениями кода</td></tr></table> Ключ с правильным ответом: А1, Б2, В3	Компонент	Назначение	А. Компилятор	1. Преобразование исходного кода в машинный	Б. Отладчик	2. Пошаговое выполнение программы	В. Система контроля версий	3. Управление изменениями кода	ОПК-8.3.1
Компонент	Назначение									
А. Компилятор	1. Преобразование исходного кода в машинный									
Б. Отладчик	2. Пошаговое выполнение программы									
В. Система контроля версий	3. Управление изменениями кода									
22	Задание закрытого типа на установление правильной последовательности Инструкция: Прочитайте текст и установите последовательность. Этапы компиляции программы: 1. Препроцессинг 2. Компиляция 3. Ассемблирование 4. Компоновка 5. Генерация исполняемого файла Запишите последовательность цифр слева направо. Ключ с правильным ответом: 1, 2, 3, 4, 5	ОПК-8.3.1								
23	Задание открытого типа Инструкция: Прочитайте текст и запишите ответ. Как называется формальное описание синтаксиса языка программирования? Ключ с правильным ответом: BNF	ОПК-8.3.1								
24	Задание открытого типа с развернутым обоснованным ответом Инструкция: Прочитайте текст и запишите развернутый обоснованный ответ	ОПК-8.3.1								

	Объясните, почему современные стандарты C++ уделяют внимание безопасности работы с памятью. Ключ с правильным ответом (эталонный ответ): Ошибки управления памятью приводят к утечкам, повреждению данных и уязвимостям безопасности. Современные стандарты C++ вводят механизмы RAII, умные указатели и строгие правила владения, что снижает риск ошибок и повышает надёжность программ.									
25	Задание закрытого типа с выбором одного правильного ответа Инструкция: Прочитайте текст и выберите один правильный ответ Какой оператор используется для разыменования указателя? 1. & 2. * 3. -> 4. sizeof Ключ с правильным ответом: 2	ОПК-8.У.1								
26	Задание закрытого типа с выбором нескольких правильных ответов Инструкция: Прочитайте текст и выберите все правильные ответы Какие инструменты применяются для отладки программ? 1. gdb 2. sanitizers 3. assert 4. linker Ключ с правильным ответом: 1, 2, 3	ОПК-8.У.1								
27	Задание закрытого типа на установление соответствия Инструкция: Прочитайте текст и установите соответствие Сопоставьте тип ошибки и пример. <table><tr><td>Тип ошибки</td><td>Пример</td></tr><tr><td>А. Логическая</td><td>1. Неверный результат при корректной компиляции</td></tr><tr><td>Б. Синтаксическая</td><td>2. Пропущенная точка с запятой</td></tr><tr><td>В. Ошибка времени выполнения</td><td>3. Деление на ноль</td></tr></table> Ключ с правильным ответом: А1, Б2, В3	Тип ошибки	Пример	А. Логическая	1. Неверный результат при корректной компиляции	Б. Синтаксическая	2. Пропущенная точка с запятой	В. Ошибка времени выполнения	3. Деление на ноль	ОПК-8.У.1
Тип ошибки	Пример									
А. Логическая	1. Неверный результат при корректной компиляции									
Б. Синтаксическая	2. Пропущенная точка с запятой									
В. Ошибка времени выполнения	3. Деление на ноль									
28	Задание закрытого типа на установление правильной последовательности Инструкция: Прочитайте текст и установите последовательность. Этапы отладки программы: 1. Воспроизведение ошибки 2. Анализ причины 3. Исправление кода 4. Повторный запуск 5. Подтверждение устранения ошибки Запишите последовательность цифр слева направо. Ключ с правильным ответом: 1, 2, 3, 4, 5	ОПК-8.У.1								
29	Задание открытого типа Инструкция: Прочитайте текст и запишите ответ. Как называется механизм передачи параметров в функцию, при котором передаётся копия значения? Ключ с правильным ответом: передача по значению	ОПК-8.У.1								
30	Задание открытого типа с развернутым обоснованным ответом Инструкция: Прочитайте текст и запишите развернутый обоснованный ответ	ОПК-8.У.1								

	Объясните, почему важно тестировать программу на граничных значениях входных данных. Ключ с правильным ответом (эталонный ответ): Граничные значения часто выявляют ошибки, связанные с переполнением, неверными условиями и некорректной обработкой крайних случаев. Тестирование таких значений повышает надёжность программы и предотвращает сбои в реальных условиях.									
31	Задание закрытого типа с выбором одного правильного ответа Инструкция: Прочитайте текст и выберите один правильный ответ Какой инструмент позволяет обнаруживать утечки памяти? 1. gdb 2. valgrind 3. CMake 4. clang-format Ключ с правильным ответом: 2	ОПК-8.В.1								
32	Задание закрытого типа с выбором нескольких правильных ответов Инструкция: Прочитайте текст и выберите все правильные ответы Какие действия относятся к процессу тестирования программы? 1. Подготовка тестов 2. Анализ результатов 3. Форматирование кода 4. Проверка корректности работы функций Ключ с правильным ответом: 1, 2, 4	ОПК-8.В.1								
33	Задание закрытого типа на установление соответствия Инструкция: Прочитайте текст и установите соответствие Сопоставьте инструмент и тип выявляемых ошибок. <table><tr><td>Инструмент</td><td>Ошибки</td></tr><tr><td>А. sanitizers</td><td>1. Ошибки работы с памятью</td></tr><tr><td>Б. valgrind</td><td>2. Утечки памяти</td></tr><tr><td>В. gdb</td><td>3. Логические и пошаговые ошибки</td></tr></table> Ключ с правильным ответом: А1, Б2, В3	Инструмент	Ошибки	А. sanitizers	1. Ошибки работы с памятью	Б. valgrind	2. Утечки памяти	В. gdb	3. Логические и пошаговые ошибки	ОПК-8.В.1
Инструмент	Ошибки									
А. sanitizers	1. Ошибки работы с памятью									
Б. valgrind	2. Утечки памяти									
В. gdb	3. Логические и пошаговые ошибки									
34	Задание закрытого типа на установление правильной последовательности Инструкция: Прочитайте текст и установите последовательность. Этапы тестирования функции: 1. Определение входных данных 2. Запуск функции 3. Сравнение результата с ожидаемым 4. Анализ ошибок 5. Повторное тестирование Запишите последовательность цифр слева направо. Ключ с правильным ответом: 1, 2, 3, 4, 5	ОПК-8.В.1								
35	Задание открытого типа Инструкция: Прочитайте текст и запишите ответ. Как называется инструмент, позволяющий выполнять программу пошагово? Ключ с правильным ответом: gdb	ОПК-8.В.1								
36	Задание открытого типа с развернутым обоснованным ответом Инструкция: Прочитайте текст и запишите развернутый обоснованный ответ Объясните, почему важно использовать инструменты анализа памяти при разработке программ.	ОПК-8.В.1								

	Ключ с правильным ответом (эталонный ответ): Инструменты анализа памяти позволяют выявлять утечки, двойное освобождение, выход за границы массива и другие ошибки, которые трудно обнаружить вручную. Это повышает надёжность и безопасность программного обеспечения, снижает риск непредсказуемого поведения и сбоев, а также упрощает сопровождение и развитие кода.	
--	---	--

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:

- постановка задачи;
- основные сведения по теме лекции;
- результаты и выводы.

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

Задания и требования к проведению лабораторных работ размещаются в личном кабинете обучающегося <https://pro.guap.ru/>.

Структура и форма отчета о лабораторной работе

Отчет о лабораторной работе сдается в электронном виде (документ Word, документ PDF) через Личный кабинет ГУАП. Отчет к лабораторной работе содержит следующие элементы:

- титульный лист с названием дисциплины, номером и названием лабораторной работы;
- цели и задачи работы;
- задание;
- решение;
- выводы.

Требования к оформлению отчета о лабораторной работе

Общие требования и рекомендации по выполнению письменных работ : методические указания (с изменениями от 09.01.2019) [Электронный ресурс] / Ивангородский филиал С.-Петерб. гос. ун-т аэрокосм. приборостроения ; сост. А. А. Сорокин. - Ивангород : 2019. - 37 с. Режим доступа: <https://ifguap.ru/rp/ReportsFormattingRules.pdf>, Личный кабинет ГУАП

11.3. Методические указания для обучающихся по выполнению курсовой работы

Курсовая работа проводится с целью формирования у обучающихся опыта комплексного решения конкретных задач профессиональной деятельности.

Курсовая работа позволяет обучающемуся:

- закрепить индикаторы достижения компетенций по дисциплине за счет самостоятельной детальной проработки темы в рамках дисциплины;
- показать освоение индикаторов достижения компетенций и знание материала дисциплины;
- подготовиться к более сложной технической задаче - выполнению выпускной квалификационной работе;
- определиться с направлением или возможной темой выпускной квалификационной работы.

Структура пояснительной записки курсовой работы

- титульный лист с названием дисциплины;
- техническое задание;
- введение
- разделы отчета
- заключение.

Требования к оформлению пояснительной записки курсовой работы

Общие требования и рекомендации по выполнению письменных работ : методические указания (с изменениями от 09.01.2019) [Электронный ресурс] / Ивангородский филиал С.-Петерб. гос. ун-т аэрокосм. приборостроения ; сост. А. А. Сорокин. - Ивангород : 2019. - 37 с. Режим доступа: <https://ifguap.ru/rp/ReportsFormattingRules.pdf>, Личный кабинет ГУАП

11.4. 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 ему развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Основными методическими материалами, направляющими самостоятельную работу обучающихся, являются источники из перечня печатных и электронных учебных изданий, указанных в таблице 8. Кроме этого, обучающийся может пользоваться электронными ресурсами, указанными в таблицах 9 и 11.

11.5. Методические указания для обучающихся по прохождению консультаций.

По изучаемой дисциплине проводятся следующие виды консультаций:

Консультация перед экзаменом - проводится с целью:

- уточнения организационных моментов;
- систематизации знаний;
- ответы на вопросы, вызывающие трудности при подготовке к экзамену.

Консультация имеет форму лекции, после которой преподаватель отвечает на вопросы студентов или в виде беседы в форме "ответ-вопрос".

Консультация со слабоуспевающими студентами - предназначена для:

- ликвидации пробелов при изучении дисциплины;
- разъяснения спорных вопросов и вопросов, наиболее сложных для изучения;
- закрепления пройденного материала;
- ликвидации академических задолженностей.

Проводится регулярно согласно графику консультаций преподавателя (не реже 1 раза в 2 недели).

Консультация по проектной и научно-исследовательской деятельности обучающихся - проводится с целью:

- расширения научного кругозора обучающихся;
- рассмотрения вопросов, не включенных в программу изучаемой дисциплины;
- углубленного изучения материала курса;
- помощи обучающимся в подготовке научных статей и докладов на конференции;
- подготовки к участию в конкурсах и олимпиадах.

Проводится регулярно согласно графику консультаций преподавателя или по устной договоренности между обучающимся и преподавателем.

11.6. 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. Невыполнение требований или их части по прохождению текущего контроля успеваемости при успешном прохождении промежуточной аттестации может привести к понижению итоговой оценки.

Возможные методы текущего контроля:

- устный опрос на занятиях;
- систематическая проверка выполнения индивидуальных и домашних заданий;
- защита отчетов по лабораторным работам;
- проведение контрольных работ;
- тестирование;
- контроль самостоятельных работ;
- проведение контрольных работ;
- выполнения контрольной работы заочников;
- контроль курсового проектирования;
- контроль выполнения курсовых работ;
- доклад на научной конференции;
- написание научной статьи.

Контроль осуществляется путем проведения опросов, проверки выполнения заданий на практических занятиях / лабораторных работах. Преподавателем осуществляется текущая оценка сформированности индикаторов ОПК 1.3.1, ОПК 1.У.1, ОПК 2.3.1, ОПК 2.У.1, ОПК 8.В.1, ОПК 8.3.1, ОПК 8.У.1. Даты проведения текущего контроля по указанным заданиям сообщаются обучающимся заранее. Студент должен выполнить все предусмотренные контрольные мероприятия не позднее указанного срока. Результаты текущего контроля учитываются при проведении промежуточной аттестации.

11.7. Методические указания для обучающихся по прохождению тестирования.

Использование тестовых заданий возможно как при текущем контроле, так и при проведении промежуточной аттестации. Тесты могут проводиться как в письменной форме, так и с использованием электронных средств обучения.

Можно выделить основные уровни теста, в которых проверка возрастает от контроля знаний (индикатор достижения компетенции - "знать") до применения навыков при решении типовых и нетиповых задач ((индикаторы достижения компетенции - "уметь" и "владеть")):

- Первый уровень - узнавание ранее изученного материала;
- Второй уровень - репродуктивный - в заданиях не содержится материала для ответа или же его извлечение требует не только запоминания материала, но и его понимания (подстановка, конструктивный тест, типовая задача);
- Третий уровень - нетиповые задачи повышенной сложности, для которых требуется самостоятельное нахождение методов решения;
- Смешанный - использование элементов всех трех уровней для проверки разных индикаторов достижения компетенций.

Критерии оценки тестовых работ базируются на 100-бальной шкале согласно Таблице 14:

- менее 51 - "не зачтено" или "неудовлетворительно" (2);
- от 51 до 69 - "зачтено" или "удовлетворительно" (3);

- от 70 до 89 - "зачтено" или "хорошо" (4);
- от 90 до 100 - "зачтено" или "отлично" (5).

11.8. 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».

Экзамен проводится в одной из следующих форм:

- экзамен проводится в устной форме в виде ответа на вопросы;
- экзамен проводится в письменной форме в виде ответа на вопросы;
- экзамен проводится в письменной форме в виде теста;
- экзамен проводится с применением средств электронного обучения (СДО

ГУАП).

В случае дистанционной формы промежуточной аттестации экзамен проводится в виде теста с применением средств электронного обучения.

Выполнение курсовой работы оценивается по 100-бальной шкале согласно МДО ГУАП. СМК 2.77 "Положение о модульно-рейтинговой системе оценки качества учебной работы студентов в ГУАП":

- менее 55 - "неудовлетворительно" (2);
- от 55 до 69 - "удовлетворительно" (3);
- от 70 до 84 - "хорошо" (4);
- от 85 до 100 - "отлично" (5).

Приблизительное распределение баллов за выполнение, оформление и защиту курсового проекта/работы приведено в таблице 20.

Таблица 20 – Приблизительное распределение баллов за выполнение курсовой работы

№ п/п	Критерий	Баллы
1	Оформление пояснительной записки соответствует требованиям	5
2	Структура пояснительной записки соответствует требованиям	5
3	КР/КП соответствует теме	5
4	Достижение целей и выполнение поставленных задач	5
5	Выполнение задания на библиографический поиск	5
6	Выполнение дополнительных требований и ограничений	10
7	Общий уровень выполнения КР/КП	15
8	Самостоятельность выполнения КР/КП	15
9	Выводы (заключение) по проделанной работе	10
10	Соблюдение допустимого объема пояснительной записки	5
11	Соблюдение выполнения сроков КР/КП *	5
12	Уровень освоения компетенций	5
13	Защита КР/КП	10
Итого		100

* Пояснительная записка сдается на проверку не позднее, чем за неделю до защиты, которая проходит на зачетной неделе.

В случае дистанционной формы промежуточной аттестации защита проводится с применением средств электронного обучения.

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