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

Кафедра № 44

УТВЕРЖДАЮ
Руководитель направления

д.т.н., проф.
(должность, уч. степень, звание)

М.Б. Сергеев
(инициалы, фамилия)

(подпись)
«07» марта 2019 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	Вычислительные машины, комплексы, системы и сети
Форма обучения	очно-заочная

Санкт-Петербург– 2019

Лист согласования рабочей программы дисциплины

Программу составил

доц., к.т.н., доц. (должность, уч. степень, звание) — 07 марта 2019 (подпись, дата) — Л.Н. Бариков (инициалы, фамилия)

Программа одобрена на заседании кафедры № 44
«07» марта 20__ г, протокол № 6-18/19

Заведующий кафедрой № 44

д.т.н., проф. (уч. степень, звание) — 07 марта 2019 (подпись, дата) — М.Б. Сергеев (инициалы, фамилия)

Ответственный за ОП ВО 09.03.01(02)

доц., к.т.н., доц. (должность, уч. степень, звание) — 07 марта 2019 (подпись, дата) — Н.В. Соловьев (инициалы, фамилия)

Заместитель директора института/ декана факультета № 4 по методической работе

доц., к.т.н., доц. (должность, уч. степень, звание) — 07 марта 2019 (подпись, дата) — А.А. Ключарев (инициалы, фамилия)

Аннотация

Дисциплина «Основы программирования» входит в образовательную программу высшего образования по направлению подготовки/ специальности 09.03.01 «Информатика и вычислительная техника» направленности «Вычислительные машины, комплексы, системы и сети». Дисциплина реализуется кафедрой «№44».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

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

ОПК-2 «Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, 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, лабораторные работы, 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

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 общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	ОПК-1.З.1 знать основы математики, физики, вычислительной техники и программирования ОПК-1.У.1 уметь решать стандартные профессиональные задачи с применением естественнонаучных и обще-инженерных знаний, методов математического анализа и моделирования ОПК-1.В.1 владеть навыками теоретического и экспериментального исследования объектов профессиональной деятельности
Общепрофессиональные компетенции	ОПК-2 Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности	ОПК-2.З.1 знать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности ОПК-2.У.1 уметь выбирать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности ОПК-2.В.1 владеть навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности
Общепрофессиональные компетенции	ОПК-8 Способен разрабатывать алгоритмы и	ОПК-8.З.1 знать алгоритмические языки программирования, операционные системы и оболочки, современные среды

	программы, пригодные для практического применения	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,
- Математическая логика и теория алгоритмов

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

- Программирование на языках Ассемблера
- Технология программирования
- Человеко-машинный интерфейс
- Базы данных
- Системное программное обеспечение
- Операционные системы
- Открытые системы
- Цифровая обработка изображений
- Теория принятия решений
- Технология разработки открытого программного обеспечения
- Мультимедиа производство,

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	4 / 144	5 / 180
Аудиторные занятия, всего час.	119	68	51
в том числе:			
лекции (Л), (час)	51	34	17
практические/семинарские занятия (ПЗ), (час)	-	-	-
лабораторные работы (ЛР), (час)	51	34	17
курсовой проект (работа) (КП, КР), (час)	17		17
экзамен, (час)	72	36	36
Самостоятельная работа, всего (час)	133	40	93
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.**)	Экз., 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) (час)	ЛР (час)	КП (час)	СРС (час)
Семестр 2					
Раздел 1. Методы разработки программных средств	6	-	2	-	5
Раздел 2. Основы языка программирования С++	6	-	-	-	5
Раздел 3. Структурное программирование	6	-	12	-	10
Раздел 4. Процедурное и модульное программирование	16	-	20	-	20
Итого в семестре:	34	-	34	-	40
Семестр 3					
Раздел 5. Динамические структуры данных	9	-	17	-	30
Раздел 6. Концепции объектно-ориентированного программирования	8	-		-	20
Выполнение курсовой работы	-	-		17	43
Итого в семестре:	17	-	17	17	93
Итого	51	0	51	17	133

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	Тема 1.1. Современные методы разработки программных средств. Развитие современных методов программирования. Структурное, процедурное, модульное и объектно-ориентированное программирование. Технологии нисходящего и восходящего проектирования программ. Сущность структурного программирования: разбиение на подзадачи, нисходящее проектирование, стандартные структуры управления. Достоинства и недостатки. Тема 1.2. Этапы разработки программных средств. Кодирование и документирование программ. Проектирование и тестирование программ. Основные этапы разработки программ. Этап постановки задачи. Этап разработки структур данных: определение входных и выходных данных, формы получения результатов. Использование технологии нисходящего проектирования на этапе проектирования программных средств. Этап структурного программирования. Этап тестирования и отладки.
2	Тема 2.1. Современные языки программирования. Понятие языка программирования. Этапы развития языков

	программирования. Современные тенденции в области языков программирования. Сравнение развития языков в представлении данных и способах реализации алгоритмов. Сравнительная характеристика языков программирования высокого уровня. Синтаксис и семантика. Способы описания синтаксиса: лингвистические формулы и синтаксические диаграммы. Тема 2.2. Базовые средства языка программирования С++. Структура языка программирования С++. Базовые элементы языка С++: алфавит, лексемы, выражения, комментарии. Основные (стандартные) типы данных языка С++. Типы арифметические целочисленные. Управляющие последовательности. Типы арифметические с плавающей точкой. Логический тип bool. Тип void. Идентификаторы языка. Область действия и область видимости идентификатора. Ключевые (зарезервированные) слова. Константы. Разделители. Спецификации классов памяти. Предложения языка: описания и операторы. Программа на языке С++: состав и структура. Тема 2.3. Понятие объекта. Объекты. Классификация объектов. Арифметические операции над объектами. Арифметические преобразования. Логические и поразрядные логические операции над объектами. Операции сдвига. Понятие об L-значении. Операции присваивания. Преобразования при присваивании. Операция запятая. Операции увеличения и уменьшения значения. Условная операция. Операция получения адреса объекта в основной памяти и операция получения содержимого объекта по адресу. Указатели. Указатели на объект, на функцию, на void. Операции с указателями. Ссылки. Операция приведения. Операция размер. Первичные операции. Понятие выражения. Порядок выполнения операций в выражении.
3	Тема 3.1. Базовые конструкции структурного программирования. Средства реализации линейных алгоритмов: оператор - выражение, составной оператор (блок), пустой оператор. Средства реализации разветвляющихся алгоритмов: условный оператор, оператор - переключатель, оператор перехода, оператор разрыва, оператор перехода на начало следующей итерации, оператор возврата в вызывающую функцию. Средства реализации циклических алгоритмов: операторы цикла с предусловием, с постусловием, с параметром. Реализация арифметических, итерационных и вложенных циклов. Реализация рекуррентных вычислений. Тема 3.2. Примеры разработки программ. Линейные программы. Вычисление результатов поразрядных логических операций над целыми числами. Разветвляющиеся программы – поиск экстремальных значений. Попадание точки в заданные области на плоскости. Циклические программы. Решение задач на заданное число повторений. Решение задач на получение

	результатов с заданной точностью. Получение таблиц значений алгебраических функций в заданном диапазоне изменения значения аргумента. 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	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. Функция main(). 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: назначение, структура, трансляция, тестирование. Особенности использования модулей. Модульные программы. Директива #include. Директива #define. Построение многомодульных программ средствами языка программирования С++.
5	Тема 5.1. Динамические структуры данных. Указатели и ссылки. Динамические массивы. Динамические структуры: списки. Виды списков: односвязные и двусвязные списки, линейные и циклические списки. Динамические структуры: стеки. Динамические структуры: очереди. Динамические структуры: деки. Тема 5.2. Деревья. Динамические структуры: деревья. Виды деревьев и способы их реализации. Бинарное дерево как рекурсивная структура данных. Рекурсивные процедуры обхода дерева: инфиксная форма, префиксная форма, постфиксная форма. Особенности использования рекурсии при построении дерева.
6	Тема 6.1. Классы. Основные принципы объектно-ориентированного программирования: инкапсуляция, наследование, полиморфизм. Классы в языке С++. Разработка классов. Составляющие классов: поля, методы. Экземпляры класса (объекты). Время их жизни и видимость. Конструкторы.

	Свойства конструкторов. Статические элементы класса (статические поля и статические методы). Дружественные функции и дружественные классы. Деструкторы. Тема 6.2 Перегрузка операций. Перегрузка операций с объектами конкретных классов. Перегрузка унарных операций. Перегрузка бинарных операций. Перегрузка операции присваивания. Перегрузка операции приведения типа. Перегрузка операции вызова функции. Перегрузка операции индексирования. Тема 6.3. Наследование Понятие наследования. Простое наследование. Виртуальные методы. Механизм позднего связывания. Множественное наследование. Шаблоны классов. Создание шаблонов классов. Использование шаблонов классов. Тема 6.4 Потоковые классы. Стандартные потоки. Методы обмена с потоками (операция извлечения из потока и операция включения в поток). Файловые потоки. Строковые потоки. Тема 6.5 Строки в стиле языка С++. Класс string. Конструкторы строк. Присваивание строк. Операции со строками. Функции добавления частей строк. Функции преобразования строк. Функции поиска подстрок. Функции сравнения частей строк. Функции получения 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)	№ раздела дисциплины
Учебным планом не предусмотрено				

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)	№ раздела дисциплины
Семестр 2			
	Вводное занятие	2	1
1	Следование	4	3
2	Операторы ветвления	4	3
3	Операторы цикла: заданное число повторений	4	3
4	Операторы цикла: вычисления с заданной точностью	4	3
5	Суммирование рядов	4	4
6	Перегрузка функций	4	4
7	Статические одномерные массивы	4	4
8	Статические двумерные массивы	4	4
Семестр 3			
9	Динамические массивы	4	5
10	Методы сортировки	3	5
11	Передача имён функций в качестве параметров	2	5
12	Строки и текстовые файлы	2	5
13	Структуры и файлы структур	2	5
14	Линейные списки	4	5
	Всего	51	

4.5. Курсовое проектирование/ выполнение курсовой работы

Цель курсовой работы: закрепление теоретических и практических знаний, полученных во время лекционных, лабораторных, практических и самостоятельных занятий.

За время курсового проектирования необходимо выполнить все этапы решения задачи с помощью ЭВМ от постановки задачи до выпуска документации на разработанное программное средство. Курсовой проект заканчивается оформлением пояснительной записки и устной защитой работы с показом работы программного средства на ЭВМ.

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)	82	36	46
Курсовое проектирование (КП, КР)	43	-	43
Подготовка к текущему контролю успеваемости (ТКУ)	8	4	4
Всего:	133	40	93

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 Б24	Бариков Л.Н. Основы алгоритмизации и программирования на С/С++ [Текст]: учебное пособие / Л.Н. Бариков,– СПб.: ГУАП, 2019. - 164с.	50
004.4 П12	Павловская Т.А. С/С++. Программирование на языке высокого уровня [Текст]: учебник / Т.А. Павловская. - СПб.: ПИТЕР, 2003. - 459с.	49

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
	Не предусмотрено

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	Borland C++ 3.1
2	Geany
3	Visual Studio Community 2015

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	Лекционная аудитория	
2	Мультимедийная лекционная аудитория	

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средствдля 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
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 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; – делает выводы и обобщения; – владеет системой специализированных понятий.

Оценка компетенции 5-балльная шкала	Характеристика сформированных компетенций
«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
	Семестр 2
1	Базовые элементы языка C++: алфавит, лексемы, выражения, комментарии.
2	Концепция данных в языке C++.
3	Основные (стандартные) типы данных языка C++.
4	Типы арифметические целочисленные. Управляющие последовательности.
5	Типы арифметические с плавающей точкой.
6	Логический тип bool. Тип void.
7	Предложения языка C++: описания и операторы.
8	Программа на языке C++: состав и структура.
9	Идентификаторы языка. Область действия и область видимости идентификатора.
10	Ключевые (зарезервированные) слова.
11	Константы. Разделители.
12	Спецификации классов памяти.
13	Объекты. Классификация объектов.
14	Арифметические операции над объектами. Арифметические преобразования.
15	Логические и поразрядные логические операции над объектами.
16	Операции сдвига. Понятие об L-значении.
17	Операции присваивания. Преобразования при присваивании.
18	Операция запятая. Операции увеличения и уменьшения значения.
19	Условная операция. Операция получения адреса объекта в основной памяти и операция получения содержимого объекта по адресу.
20	Указатели. Указатели на объект, на функцию, на void. Операции с указателями.
21	Ссылки.
22	Операция приведения. Операция размер. Первичные операции.
23	Понятие выражения. Порядок выполнения операций в выражении.
24	Средства реализации линейных алгоритмов: оператор - выражение, составной оператор (блок), пустой оператор.
25	Средства реализации разветвляющихся алгоритмов: условный оператор, оператор - переключатель, оператор перехода, оператор разрыва, оператор перехода на начало следующей итерации, оператор возврата в вызывающую функцию.
26	Средства реализации циклических алгоритмов: операторы цикла с предусловием, с постусловием, с параметром.
27	Реализация арифметических, итерационных и вложенных циклов.
28	Реализация рекуррентных вычислений.
29	Директивы препроцессора: директива #include.
30	Директивы препроцессора: директива #define.
	Семестр 3
1	Структурированные типы данных: массивы одномерные и многомерные.
2	Строки.
3	Структурированные типы данных: структуры (struct).
4	Структурированные типы данных: перечисления (enum).
5	Структурированные типы данных: объединения (union).
6	Переименование типов (typedef).
7	Статическое распределение памяти.
8	Функции. Основные понятия. Принципы использования функций в программах.
9	Объявление и определение функций.
10	Список параметров и тип функции.
11	Вызов функций на исполнение. Формальные и фактические параметры.
12	Области действия описаний функций. Локальность и глобальность.
13	Механизм передачи параметров.
14	Глобальные переменные.
15	Функция main().
16	Передача массивов в качестве параметров. Примеры.
17	Реализация алгоритмов сортировки структур данных и поиска в этих структурах.
18	Передача имен функций в качестве параметров.
19	Понятие рекурсии. Рекурсивные определения и алгоритмы.
20	Программирование рекурсивных алгоритмов: рекурсивные функции. Механизм рекурсивных вызовов.
21	Перегрузка функций.
22	Шаблоны функций.
23	Динамические структуры: списки. Виды списков: односвязные и двусвязные списки, линейные и циклические списки.
24	Динамические структуры: стеки.
25	Динамические структуры: очереди.
26	Динамические структуры: деки.
27	Динамические структуры: деревья. Виды деревьев и способы их реализации.
28	Классы в языке C++.
29	Разработка классов. Составляющие классов: поля, методы.
30	Экземпляры класса (объекты). Время их жизни и видимость.
31	Конструкторы. Свойства конструкторов.
32	Статические элементы класса (статические поля и статические методы).

33	Наследование. Простое наследование.
34	Виртуальные методы. Механизм позднего связывания.
35	Критерии качества программы.
36	Жизненный цикл программы.

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
	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
	Решение задач линейного программирования:
1	Транспортная задача.
2	Задача о назначениях.
3	Задача об оптимальном ассортименте.
4	Задача о диете.
5	Задача о максимальном потоке.
	Решение задач теории расписаний:
6	Задача одного исполнителя на минимакс.
7	Задача одного исполнителя на минисумму.
8	Задача о нескольких исполнителях.
	Решение прикладных задач теории графов:
9	Задача о дорожной сети.
10	Задача о телефонной связи.
11	Задача строительной трассировки.
12	Задача электронной трассировки.
13	Задача размещения.
	Реализация алгоритмов теории графов:
14	Поиск в глубину графа.
15	Поиск в ширину графа.
16	Кратчайшие пути в графе.
17	Эйлеровы пути.
18	Гамильтоновы циклы.
	Разработка программ создания и использования баз данных в различных предметных областях:
19	База данных телефонной сети.
20	База данных библиотеки.
21	База данных магазина грамзаписей.
	Разработка программ разного назначения:
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
	Не предусмотрено

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

Целью дисциплины является – получение студентами необходимых знаний, умений и навыков 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.

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 материала по рассматриваемой теме;

– демонстрация примеров составления схем алгоритмов решения конкретных задач;

– ответы на возникающие вопросы по теме лекции;

– выдача раздаточного материала с примерами программ по теме лекции и обсуждение их особенностей.

Структура лекционного материала приведена в издании:

Бариков Л.Н. Основы алгоритмизации и программирования на С/С++ [Текст]: учебное пособие / Л.Н. Бариков,– СПб.: ГУАП, 2019. - 164с.

11.2. Методические указания для обучающихся по выполнению лабораторных работ *(если предусмотрено учебным планом по данной дисциплине)*

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

По каждой лабораторной работе обучающийся получает вариант индивидуального задания в соответствии с его номером в списке группы. Перед проведением лабораторной работы обучающемуся следует внимательно ознакомиться с методическими указаниями по её выполнению. В соответствии с заданием обучающийся должен выполнить постановку задачи: сформулировать условие, определить входные и выходные данные, разработать математическую модель. После этого он должен построить схему алгоритма решения задачи и защитить её у преподавателя. Это является допуском к реализации алгоритма на компьютере. После отладки программы обучающийся должен продемонстрировать преподавателю работу программы на полном наборе тестов. Лабораторная работа завершается оформлением и защитой отчета по лабораторной работе.

Структура и форма отчета о лабораторной работе

Отчет о лабораторной работе должен включать в себя: титульный лист, формулировку задания, математическая модель, схема алгоритма решения задачи, текст программы, контрольные (тестовые) примеры.

Требования к оформлению отчета о лабораторной работе

По каждой лабораторной работе выполняется отдельный отчет. Титульный лист оформляется в соответствии с шаблоном (образцом) приведенным на сайте ГУАП (www.guap.ru) в разделе «Сектор нормативной документации». Текстовые и графические материалы оформляются в соответствии с действующими ГОСТами и требованиями, приведенными на сайте ГУАП (www.guap.ru) в разделе «Сектор нормативной документации».

Методические указания к выполнению лабораторных работ включены в состав следующих изданий:

Бариков Л.Н. Основы алгоритмизации и программирования на С/С++ [Текст]: учебное пособие / Л.Н. Бариков,– СПб.: ГУАП, 2019. - 164с.

11.3. Методические указания для обучающихся по прохождению курсового проектирования/выполнения курсовой 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 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.

За время курсового проектирования необходимо выполнить все этапы решения задачи с помощью ЭВМ от постановки задачи до выпуска документации на разработанное программное средство. Курсовой проект заканчивается оформлением пояснительной записки и устной защитой работы с показом работы программного средства на ЭВМ.

Структура пояснительной записки курсового проекта/ работы

В состав пояснительной записки должны входить:

- титульный лист;
- содержание пояснительной записки;
- введение;
- четыре программных документа, оформленных по Стандартам ЕСПД:
  - Спецификация.
  - Текст программы.
  - Описание программы.
  - Описание применения.
- заключение;
- список использованных источников.

Руководитель курсового проектирования может в каждом конкретном случае потребовать дополнительного оформления еще ряда документов.

Требования к оформлению пояснительной записки курсового проекта/ работы

Документ **Пояснительная записка** содержит укрупненные схемы алгоритмов, общие описания алгоритмов и функционирования программы, а также обоснование принятых технических и технико-экономических решений. Требования к содержанию и оформлению **Пояснительной записки** устанавливает ГОСТ 19.404-79.

Требования к содержанию и оформлению документа **Спецификация** устанавливает ГОСТ 19.202-78. Этот ГОСТ определяет программный документ **Спецификация** как состав программы и документации на неё.

Требования к содержанию и оформлению документа **Текст программы** устанавливает ГОСТ 19.401-78. Этот ГОСТ определяет программный документ **Текст программы** как запись программы с необходимыми комментариями.

Требования к содержанию и оформлению документа **Описание программы** устанавливает ГОСТ 19.402-78. Этот ГОСТ определяет программный документ **Описание программы** как совокупность сведений о логической структуре и функционировании программы.

Документ **Описание применения** содержит сведения о назначении программы, области применения, применяемых методах, класса решаемых задач, ограничениях для применения, минимальной конфигурации технических средств, операционной среде. Требования к содержанию и оформлению этого программного документа устанавливает ГОСТ 19.502-78.

Методические указания к выполнению курсовой работы / проекта:

Бариков Л.Н. Документирование программного обеспечения. [Текст]: Методические указания к выполнению курсовой работы / Л.Н. Бариков. – СПб.: ГУАП, 2013. – 24 с.

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.

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-методические материалы по дисциплине.

11.5. 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.

Такой контроль включает в себя защиту отчетов по всем выполненным лабораторным работам. В ходе защиты обучающиеся обосновывают выбранную математическую модель решения задачи и схемы алгоритмов ее решения. Каждая работа оценивается исходя из максимального количества рейтинговых баллов, предусмотренных за эту работу.

Сумма полученных рейтинговых баллов за выполненные лабораторные работы учитывается при проведении промежуточной аттестации (экзамена).

11.6. 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».

Система оценок при проведении промежуточной аттестации осуществляется в соответствии с требованиями Положений «О текущем контроле успеваемости и промежуточной аттестации студентов ГУАП, обучающихся по программе 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