

федеральное государственное бюджетное образовательное учреждение
высшего образования
РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ
УНИВЕРСИТЕТ

Кафедра Информационных технологий и систем безопасности

Рабочая программа по дисциплине

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

Основная профессиональная образовательная программа
высшего образования программы специалитета по специальности

10.05.02 «Информационная безопасность телекоммуникационных систем»

Специализация:

Разработка защищенных телекоммуникационных систем

Квалификация:

Специалист

Форма обучения

Очная

Согласовано
Руководитель ОПОП
«Информационная безопасность
телекоммуникационных систем»


Бурлов В.Г.

Утверждаю

Председатель УМС  И.И. Палкин

Рекомендована решением
Учебно-методического совета
«11» июня 2019 г., протокол № 7

Рассмотрена и утверждена на заседании кафедры
«07» мая 2019 г., протокол № 5

Зав. кафедрой  Завгородний В.Н.

Авторы-разработчики:


Богданов П.Ю.

**федеральное государственное бюджетное образовательное учреждение
высшего образования
РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ
УНИВЕРСИТЕТ**

Кафедра Информационных технологий и систем безопасности

Рабочая программа по дисциплине

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

Основная профессиональная образовательная программа
высшего образования программы специалитета по специальности

10.05.02 «Информационная безопасность телекоммуникационных систем»

Специализация:

Разработка защищенных телекоммуникационных систем

Квалификация:

Специалист

Форма обучения

Очная

Согласовано
Руководитель ОПОП
«Информационная безопасность
телекоммуникационных систем»

_____ Бурлов В.Г.

Утверждаю

Председатель УМС _____ И.И. Палкин

Рекомендована решением

Учебно-методического совета

«__» _____ 2018 г., протокол № ____

Рассмотрена и утверждена на заседании кафедры

_____ 2018 г., протокол № ____

Зав. кафедрой _____ Бурлов В.Г.

Авторы-разработчики:

_____ Шишкин А.Д.

Санкт-Петербург 2018

1. Цели освоения дисциплины

Цель дисциплины:

- является формирование общей информационной культуры студентов, подготовка их к деятельности, связанной с использованием современных информационных технологий;
- подготовка специалистов к деятельности, связанной с разработкой программного обеспечения для решения профессиональных задач;
- формирование теоретических знаний, освоение методов и средств программирования практических инженерных задач на современных пакетах объектно-ориентированного языка, подготовка специалистов к деятельности, связанной с разработкой и эффективной реализацией программных средств для решения профессиональных задач

Основные задачи дисциплины:

- изучение основных понятий информатики;
- изучение свойств и способов записи алгоритмов;
- изучение способов представления чисел, символов, графики, аудио- и видеоинформации в персональном компьютере;
- ознакомление с логическими основами устройства ЭВМ;
- ознакомление с составом и назначением функциональных узлов компьютера;
- изучение основ построения операционных систем (ОС) на примере ОС с открытым кодом;
- изучение основ программирования в командных оболочках;
- овладение навыками применения сервисных программных средств системного и прикладного назначения;
- изучение основ построения компьютерных сетей;
- овладение навыками поиска информации в глобальной информационной сети Интернет;
- ознакомление с теоретическими основами программирования;
- изучение основ алгоритмизации;
- изучение средств описания данных и средств описания действий языков программирования;
- овладение навыками и приемами программирования;
- освоение современных сред создания программных продуктов
- изучение базовых структур данных;
- изучение основных алгоритмов сортировки и поиска;
- изучение алгоритмов генерации перестановок, подмножеств;
- изучение основных алгоритмов на графах и паттернов программирования;
- освоение основных методов оценки вычислительной сложности алгоритмов.

– –обучить технологии составления проектно-технической документации на разрабатываемый программный продукт.

2. Место дисциплины в структуре ОП

Дисциплина «Информатика и программирование» читается студентам 1 курса и относится к числу дисциплин базовой части Блока1 Дисциплины (Модули).

. Для успешного усвоения дисциплины «Информатика и программирование» необходимо, чтобы студент владел знаниями, умениями и навыками, сформированными в школе при изучении: математики, английского языка и информатики.

Параллельно с данной дисциплиной идет изучение таких взаимосвязанных с «Информатикой и программированием» с дисциплин как:

- «Физика»
- «Линейная алгебра»
- «Аналитическая геометрия»
- «Математический анализ»
- «Дискретная математика»
- «Иностранный язык»
- «
- «Аппаратные средства вычислительной техники»

Дисциплина «Информатика и программирование» обеспечивает изучение следующих дисциплин:

- Интернет-технологии»
- «Информационные технологии» «Операционные системы»
- «Моделирование систем и сетей телекоммуникаций»;
- «Геоинформационные технологии в телекоммуникационных системах»
- «Сетевое администрирование»
- «Защищенные корпоративные сети»

Дисциплина «Информатика и программирование» необходима для подготовка курсовых работ и дипломного проектирования

3. Компетенции обучающегося, формируемые в результате освоения дисциплины.

Процесс изучения дисциплины направлен на формирование следующих компетенций:

Код компетенции	Компетенция
ОПК-4	способностью понимать значение информации в развитии современного общества, применять достижения информационных технологий для поиска и обработки информации
ОПК-5	способностью применять программные средства системного и прикладного назначения, языки, методы и инструментальные средства программирования для решения профессиональных задач

В результате освоения компетенций в рамках дисциплины обучающийся должен:

Код компетенции	Результаты обучения
ОПК-4	<p>Знать:</p> <ul style="list-style-type: none"> – основные понятия информатики; – формы и способы представления данных в персональном компьютере; – состав, назначение функциональных компонентов и программного обеспечения персонального компьютера; – классификацию современных компьютерных систем; – типовые структуры и принципы организации компьютерных сетей <p>Уметь:</p> <ul style="list-style-type: none"> – применять типовые программные средства сервисного назначения (средства восстановления системы после сбоев, дефрагментации и очистки диска); – пользоваться сетевыми средствами для обмена данными, в том числе с использованием глобальной информационной сети Интернет; – работать с интегрированной средой разработки программного обеспечения <p>Владеть:</p> <ul style="list-style-type: none"> – навыками работы с офисными приложениями (текстовыми процессорами, электронными таблицами, средствами подготовки презентационных материалов)
ОПК-5	<p>Знать:</p> <ul style="list-style-type: none"> – основы и стандарты разработки алгоритмов вычислительных процессов; – методы составления алгоритмов; – методы формализации алгоритмов; – основные приемы реализации алгоритмов на языках структурного программирования; – основные приемы программирования, о бласти и особенности применения языков объектно-ориентированного программирования (ООП). – паттерны проектирования; – алгоритмы сортировки и поиска данных; – методы тестирования и отладки ПО. <p>Уметь:</p> <ul style="list-style-type: none"> – реализовывать на языке высокого уровня алгоритмы решения профессиональных задач, в том числе задач обработки потоков; – работать с интегрированной средой разработки программного обеспечения; – реализовывать на языке высокого уровня алгоритмы решения профессиональных задач, в том числе задач обработки потоков. <p>Владеть:</p> <ul style="list-style-type: none"> – навыками обеспечения безопасности информации с помощью типовых программных средств (антивирусов, архиваторов, стандартных сетевых средств обмена информацией). – навыками разработки, документирования, тестирования и отладки программ

Основные признаки проявленности формируемых компетенций в результате освоения дисциплины «Информатика и программирование» сведены в таблице.

Соответствие уровней освоения компетенции планируемым результатам обучения и критериям их оценивания

Уровень освоения компетенции	Результат обучения	Результат обучения
	ОПК-4: Знать, уметь, владеть	ОПК-5: Знать, уметь, владеть
минимальный	Способен дать собственную критическую оценку изучаемого материала	Владеет основными навыками работы с источниками и критической литературой
	Может соотнести основные идеи с современными проблемами	Способен представить ключевую проблему в ее связи с другими процессами
	Способен выделить характерный авторский подход	Понимает специфику основных рабочих категорий
базовый	Способен сравнивать концепции, аргументированно излагает материал	Свободно излагает материал, однако не демонстрирует навыков сравнения основных идей и концепций
	Аргументированно проводит сравнение концепций по заданной проблематике	Способен выделить и сравнить концепции, но испытывает сложности с их практической привязкой
	Способен выделить специфику концепций в заданной проблемной области	Знает основные отличия концепций в заданной проблемной области
продвинутый	Способен грамотно обосновать собственную позицию относительно решения современных проблем в заданной области	Видит источники современных проблем в заданной области анализа, владеет подходами к их решению
	Свободно ориентируется в заданной области анализа. Понимает ее основания и умеет выделить практическое значение заданной области	Выявляет основания заданной области анализа, понимает ее практическую ценность, однако испытывает затруднения в описании сложных объектов анализа
	Может дать критический анализ современным проблемам в заданной области анализа	Знает основное содержание современных научных идей в рабочей области анализа, способен их сопоставить

Соответствие уровней освоения компетенции планируемым результатам обучения и критериям их оценивания

Этап (уровень) освоения компетенции	Основные признаки проявленности компетенции (дескрипторное описание уровня)				
	1.	2.	3.	4.	5.
минимальный	не владеет	слабо ориентируется в терминологии и содержании	Способен выделить основные идеи текста, работает с критической литературой	Владеет основными навыками работы с источниками и критической литературой	Способен дать собственную критическую оценку изучаемого материала
	не умеет	не выделяет основные идеи	Способен показать основную идею в развитии	Способен представить ключевую проблему в ее связи с другими процессами	Может соотнести основные идеи с современными проблемами
	не знает	допускает грубые ошибки	Знает основные рабочие категории, однако не ориентируется в их специфике	Понимает специфику основных рабочих категорий	Способен выделить характерный авторский подход
базовый	не владеет	плохо ориентируется в терминологии и содержании	Владеет приемами поиска и систематизации, но не способен свободно изложить материал	Свободно излагает материал, однако не демонстрирует навыков сравнения основных идей и концепций	Способен сравнивать концепции, аргументированно излагает материал
	не умеет	выделяет основные идеи, но не видит проблем	Выделяет конкретную проблему, однако излишне упрощает ее	Способен выделить и сравнить концепции, но испытывает сложности с их практической привязкой	Аргументированно проводит сравнение концепций по заданной проблематике
	не знает	допускает много ошибок	Может изложить основные рабочие категории	Знает основные отличия концепций в заданной проблемной области	Способен выделить специфику концепций в заданной проблемной области
продвинутый	не владеет	ориентируется в терминологии и содержании	В общих чертах понимает основную идею, однако плохо связывает ее с существующей проблематикой	Видит источники современных проблем в заданной области анализа, владеет подходами к их решению	Способен грамотно обосновать собственную позицию относительно решения современных проблем в заданной области
	не умеет	выделяет основные идеи, но не видит их в развитии	Может понять практическое назначение основной идеи, но затрудняется выявить ее основания	Выявляет основания заданной области анализа, понимает ее практическую ценность, однако испытывает затруднения в описании сложных объектов анализа	Свободно ориентируется в заданной области анализа. Понимает ее основания и умеет выделить практическое значение заданной области
	не знает	допускает ошибки при выделении рабочей области анализа	Способен изложить основное содержание современных научных идей в рабочей области анализа	Знает основное содержание современных научных идей в рабочей области анализа, способен их сопоставить	Может дать критический анализ современным проблемам в заданной области анализа

4. Структура и содержание дисциплины

Общая трудоемкость (объем) дисциплины (модуля) составляет 16 зачетных единиц (ЗЕ*), 576 академических часов.

Год набора: 2013, 2014, 2015, 2016, 2017, 2018

Объём дисциплины	Всего часов	Часов по семестрам	
		1 семестр	2 семестр
Общая трудоёмкость дисциплины	288	144	144
Контактная работа обучающихся с преподавателям (по видам аудиторных учебных занятий) – всего:	136	72	64
в том числе:			
лекции	68	36	32
Лабораторные занятия	68	36	32
Самостоятельная работа (СРС) – всего:	152	72	80
в том числе:			
курсовая работа			
Вид промежуточной аттестации (зачет/экзамен)	Экзамен	Экзамен	Экзамен

4.1. Структура дисциплины.

Год набора: 2019

№ п/п	Раздел и тема дисциплины	Семестр	Виды учебной работы, в т.ч. самостоятельная работа студентов, час.			Формы текущего контроля успеваемости	Занятия в активной и интерактивной форме, час.	Формируемые компетенции
			Лекции	Лабораторные работы	Самост. работа			
1	Информатика	1	12	12	40	Тест, Защита лабораторных работ	24/12	ОПК-4 ОПК- 5
2	Языки программирования	1, 2	36	36	52	Тест, Защита лабораторных работ	72/36	ОПК-4 ОПК- 5
3	Методы программирования	2	20	20	60	Тест, Защита лабораторных работ	40/20	ОПК-4 ОПК- 5
Итого			68	68	152		136/68	

4.2. Содержание разделов дисциплины

4.2.1. Информатика

Информатика - предмет и задачи

Появление и развитие информатики. Структура информатики. Информация и ее свойства. Информация и данные. Меры информации. Количественная оценка представления информации и данных.

Техническая база информационных технологий

История возникновения вычислительной техники. Классификация современной вычислительной техники по назначению, принципу действия.

(ПЭВМ, СуперЭВМ, серверы). Сравнительные характеристики современных ЭВМ. Тенденции развития ЭВМ и сетей ЭВМ.

Архитектура ЭВМ

Функционально-структурная организация персональной ЭВМ (ПЭВМ). Основные характеристики ПЭВМ. Структура, назначение и конструктивное исполнение основных блоков и устройств ПЭВМ. Устройства памяти, ввода и вывода информации. Логика работы персонального компьютера.

Представление информации в ЭВМ

Арифметико-логические основы вычислительной техники. Системы счисления и формы представления чисел. Разрядность и форматы чисел. Правила арифметических действий в системах счисления. Правила перевода чисел из одной системы в другую. Форматы представления данных в ЭВМ. Кодирование чисел

Основы логики и логические основы работы компьютера

Формы мышления: понятие, суждение, умозаключение. Алгебра высказываний. Конъюнкция, дизъюнкция, инверсия. Логические выражения и таблицы истинности. Логические функции. Логические законы и правила преобразования логических выражений. Логические основы работы устройств компьютера (сумматор, триггер). Структура и виды команд компьютера

Программное обеспечение ПЭВМ

Классификация программного обеспечения ПЭВМ. Системное программное обеспечение ПЭВМ. Современные операционные системы ПЭВМ. Файловые системы. Создание и использование командного файла. Командный файл автонастройки. Программы-оболочки.

Операционные среды

Назначение и возможности операционной среды Windows. Характеристика основных режимов работы. Структура и элементы интерфейса. Назначение и возможности операционной среды UNIX. Характеристика основных режимов работы. Структура и элементы интерфейса.

Текстовый процессор

Основные понятия и базовые возможности. Копирование, перемещение и удаление текста. Форматирование текста. Работа с окнами. Особенности издательских систем. Основы создания документа.

Антивирусные программные средства

Характеристика компьютерных вирусов. Программы обнаружения и защиты от вирусов.

Архивация и компрессия информации

Общие сведения об архивации файлов. Коэффициент сжатия. Основные программы – архиваторы: основные возможности, технология работы. Многофункциональный интегрированный архиватор WinRar. Многофункциональный интегрированный архиватор WinZip.

Технология работы с электронной таблицей

История появления и развития электронной таблицы. Интерфейс электронной таблицы. Окно, рабочая книга, лист. Форматирование числовых данных. Относительная и абсолютная адресация. Графические возможности Excel.

Базы данных. Система управления базой данных (СУБД)

Базы данных: понятие, классификация, структурные элементы. Модели данных: иерархическая, сетевая, реляционная. Архитектура СУБД. Функциональные возможности СУБД. Технологии работы в СУБД.

Компьютерные сети

Назначение и классификация компьютерных сетей (КС). Распределенная обработка данных. Обобщенная структура КС. Абоненты сети. Процессы передачи данных в КС. Основные технические устройства сети (модемы, мультиплексоры, концентраторы, мосты). Основные формы взаимодействия абонентов сети. Локальные вычислительные сети, Интернет, электронная почта., WWW. Эталонные модели взаимодействия систем в КС. Стандарты протоколов в КС. Основы защиты сетевых ресурсов

Методология создания программного продукта

Методы проектирования алгоритмов и программ (функционально-ориентированные, структурирования данных, информационные модели). Объектно-ориентированные технологии. Формализация процесса программирования. Алгоритм и его свойства. Правила составления алгоритмов. Графическое представление схем алгоритмов.

4.2.2. Языки программирования

Введение

Начальные сведения о языках программирования. Роль языков программирования. Определение целей создания программы. Характеристики и свойства языков программирования. История развития языков программирования. Поколения языков.

Особенности языка Си

Этапы разработки программы. Структурные особенности языка Си.

Основные элементы языка программирования. Инструментальные средства разработки программ на языках высокого уровня в различных операционных системах. Ключевые слова и лексемы

Базовые понятия языка: Основные операции и конструкции

Алфавит, идентификаторы, служебные слова, константы, комментарии. Структура программы. Этапы создания исполняемой программы. Типизация языка. Определение типа. Контроль типов. Уровни типизации. Эквивалентность типов. Простые типы данных. Переменные. Классы памяти. Операции, выражения, операторы.

Операторы условия. Операторы организации циклов. Вложенные циклы. Операторы передачи управления.

Массивы, функции пользователя

Указатели. Ссылки. Одномерные и многомерные массивы. Строки. Связь указателей и массивов. Способы инициализации числовых и текстовых массивов. Базовые алгоритмы и программы для обработки одномерных и двумерных массивов. Динамические массивы.

Шаблоны функций. Определение и вызов функций пользователей. Способы передачи параметров в функцию. Рекурсивные функции. Подставляемые функции. Перегрузка функций. Указатели на функции. Аргументы по умолчанию. Передача массивов в функции.

Типы данных, определяемые пользователем

Понятие структуры, примеры использования структур, операции над структурами. Объединения, перечислимый тип. Определение размера памяти в байтах (операция sizeof). Динамическое выделение памяти. Динамические структуры.

Объединения, поля битов, доступ к отдельному биту.

Препроцессорные средства, Обработка файлов

Стадии препроцессорной обработки. Директивы. Макроподстановки. Условная компиляция. Препроцессоры и директивы условной компиляции.

Понятие потока. Открытие и закрытие файлов. Операции, функции ввода-вывода результатов вычислений и текстов.

Классы памяти и области видимости данных

Классы памяти. Описатели классов памяти. Модели памяти. Распределение памяти при работе программы. Управление экраном и памятью в текстовом режиме.

Динамическое распределение памяти

Функции для динамического распределения памяти. Понятие очереди, Линейные списки, Двусвязные списки. Операции над списками. Методы хранения списков. Стеки и их обработка.

Расширенное представление данных

Абстрактные типы данных. Получение абстрактного типа. Создание интерфейса. Использование интерфейса. Реализация интерфейса представления данных. Имитация реальных ситуаций с помощью очереди. Деревья двоичного поиска

Основные принципы объектно-ориентированного программирования

Основные понятия объектно-ориентированного программирования (ООП). Отличия ООП от процедурного программирования. Понятие класса и объекта. Определение шаблона класса. Использование шаблона класса. Правила описания шаблонов. Специализация шаблона класса.

Понятия наследования и полиморфизма.

Технологии объектно-ориентированного программирования

Концепция и основные правила перегрузки операций. Перегрузка операций методами класса, дружественными и обычными функциями. Перегрузка различных операций.

Наследование

Механизм наследования. Базовые и производные классы. Типы наследования. Правила наследования различных методов. Множественное наследование.

Полиморфизм

Виртуальные и чистые виртуальные функции. Абстрактные классы. Статическое и динамическое связывание. Механизм динамического связывания. Виртуальные деструкторы.

Объекты и классы

Спецификаторы доступа к членам класса. Объявление или спецификация класса. Доступ к членам класса. Типы конструкторов. Вызов конструкторов. Статистические данные класса. Константные методы класса. Конструкторы и

деструкторы. Дружественные классы и функции. Статические элементы класса. Константные методы и объекты. Указатель this. Указатели и ссылки на объекты, массивы объектов.

Потоки данных

Классификация способов ввода – вывода данных. Принципы работы с потоками и файлами. Стандартные классы потоков. Форматированный ввод-вывод базовых типов. Манипуляторы ввода и вывода данных. Файловые потоки данных.

4.2.3 Методы программирования

Введение

Задачи дисциплины, связь с другими курсами.

Проектирование программного обеспечения при структурном подходе

Разработка структурных и функциональных схем. Использование метода пошаговой детализации для проектирования структуры программного обеспечения (ПО). Структурные карты Константайна. Проектирование структур данных. Проектирование ПО, основанное на декомпозиции данных. Case-технологии, основанные на методологиях анализа и проектирования. Программирование с защитой от ошибок.

Проектирование программного обеспечения при объектно-ориентированном подходе

Объектная модель. Основные принципы и элементы объектной модели. Виды связей между элементами объектной модели. Понятие и цель объектно-ориентированного анализа. Подходы к анализу объектно-ориентированных систем. Преимущества и риски, связанные с объектно-ориентированным проектированием (ООП). Универсальный язык ООП Java. Определение и история создания языка Java. Платформа Java. Java-технология. Пакеты- Java. Методы – Java. Объявление и вызов методов. Использование строк. Основные базовые конструкции. Коллекции. Создание конструкторов. Классы, методы класса. Сравнение и перегрузка объектов. Обработка исключений. Java-приложения.

Жизненный цикл и этапы разработки программного обеспечения

Понятие технологии программирования и основные этапы ее развития. Жизненный цикл программного обеспечения (ПО). Стандарты, регламентирующие жизненный цикл. Модели жизненного цикла ПО. Основные этапы разработки сложных программных систем. Планирование и организация разработки ПО. Проектирование ПО. Кодирование ПО.

Паттерны проектирования

Понятие и описание паттерна проектирования. Принципы применения паттернов. Классификация паттернов. Подходы к выбору паттернов.

Базовые структуры данных. Алгоритмы

Базовые структуры данных: множество, стек, очередь, дек, список, деревья, двоичные деревья поиска. Понятие алгоритма. Принципы анализа алгоритмов.

Алгоритмы сортировки и поиска

Основные понятия и классификация алгоритмов сортировки. Алгоритмы внутренней сортировки: Шелла, быстрая, пирамидальная, слиянием и

лексикографическая. Внешняя сортировка: основные понятия, построение цепочек, многофазное слияние.

Основные понятия. Поиск в последовательно организованных структурах. Поиск в древовидных структурах. Хеширование и разрешение коллизий.

Алгоритмы на графах

Основные понятия. Способы задания графов. Алгоритмы обхода графов в ширину и глубину. Основные деревья минимальной стоимости и алгоритмы их построения. Алгоритмы нахождения кратчайших расстояний между вершинами.

Тестирование программных продуктов

Виды контроля качества разрабатываемого ПО. Ручной контроль. Структурное тестирование. Функциональное тестирование. Тестирование модулей и комплексное тестирование. Оценочное тестирование.

Отладка программного обеспечения

Качество ПО. Методы контроля качества ПО. Ошибки в программах. Методы отладки ПО. Документирование ПО. Сопровождение ПО.

4.3. Лабораторные занятия, их содержание

№ п/п	№ раздела дисциплины	Тема занятия	Форма проведения	Формируемые компетенции
	1	Создание однотипных документов в msword	Лабораторная	ОПК-4, ОПК-5
	1	Форматирование документов в msword	Лабораторная	ОПК-4, ОПК-5
	1	Работа с таблицами в msword	Лабораторная	ОПК-4, ОПК-5
	1	Работа со слиянием документов в msword	Лабораторная	ОПК-4, ОПК-5
	1	Организация вычислений в электронной таблице msexel	Лабораторная	ОПК-4, ОПК-5
	1	Решение квадратных уравнений систем линейных алгебраических уравнений(слау) средствами ms excel	Лабораторная	ОПК-4, ОПК-5
	1	Программирование алгоритмов линейной структуры	Лабораторная	ОПК-4, ОПК-5
	1	Программирование алгоритмов разветвляющейся структуры	Лабораторная	ОПК-4, ОПК-5
	1	Программирование алгоритмов циклической структуры	Лабораторная	ОПК-4, ОПК-5
	2	Освоение работы в интегрированной среде программирования	Лабораторная	ОПК-4, ОПК-5
	2	Базовые операции языка Си	Лабораторная	ОПК-4, ОПК-5
	2	Организация вычислительных процессов	Лабораторная	ОПК-4, ОПК-5
	2	Условные операторы и операторы выбора	Лабораторная	ОПК-4, ОПК-5
	2	Обработка одномерных массивов	Лабораторная	ОПК-4, ОПК-5
	2	Обработка двумерных массивов	Лабораторная	ОПК-4, ОПК-5

	2	Функции пользователя и динамическое распределение памяти	Лабораторная	ОПК-4, ОПК-5
	3	Организация работы с файлами	Лабораторная	ОПК-4, ОПК-5
	3	Обработка списков	Лабораторная	ОПК-4, ОПК-5
	3	Вывод графиков функций	Лабораторная	ОПК-4, ОПК-5
	3	Основы работы в среде С++	Лабораторная	ОПК-4, ОПК-5
	3	Обработка списков в С++	Лабораторная	ОПК-4, ОПК-5
	3	Вывод графиков функций в С++	Лабораторная	ОПК-4, ОПК-5
	3	Работа в среде С++	Лабораторная	ОПК-4, ОПК-5

5. Учебно-методическое обеспечение самостоятельной работы студента и оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

5.1. Текущий контроль

Текущий контроль производится путем защиты лабораторных работ в каждом семестре.

В конце каждого семестра проводится тестирование по изученным темам дисциплины. Сами тесты представлены в Системе управления курсами РГГМУ (<http://moodle.rshu.ru/>)

Критерии оценивания лабораторных работ.

- оценка «зачтено»: работа полностью выполнена. Даны полные ответы на вопросы по теме работы;
- оценка «не зачтено»: работа не выполнена или при защите студент не может ясно и четко ответить на поставленные вопросы.

5.2. Методические указания по организации самостоятельной работы

Самостоятельная работа связана с изучением и конспектированием отдельных вопросов лекционного материала, выделенного преподавателем. Для успешного выполнения самостоятельной работы необходимо:

- в соответствии с заданной темой проработать соответствующий лекционный материал;
- прочитать литературу из рекомендованного списка;
- при необходимости осуществить поиск нужной информации в сети.

Контроль выполнения самостоятельной работы обучающегося осуществляется собеседованием по определению понимания изученного материала.

Контроль исполнения самостоятельных работ осуществляется преподавателем с участием студентов в форме обсуждения выполненных заданий.

5.3. Промежуточный контроль: экзамен (1 и 2 семестры)

Перечень вопросов для промежуточной аттестации – экзамен (Раздел 1, 1 семестр):

1. Информатика, предмет и задачи; определения, структура

2. Информация и ее свойства, адекватность информации
3. Меры информации
4. Количественные меры информации и связь их с вероятностью событий.
5. История развития ЭВМ. Основные поколения.
6. Виды классификации ЭВМ
7. Супер ЭВМ: назначение, принципы обработки информации.
8. Функционально-структурная организация ПК.
9. Основные структурные элементы ПК.
10. Микропроцессоры ПК: Основные поколения
11. Микропроцессоры ПК: основные параметры и характеристики
12. Внутренняя память ПК
13. Внешняя память ПК.
14. Логическая структура и информационная емкость диска.
15. Системы счисления, обобщенная форма перевода чисел
16. Арифметические основы перевода чисел из десятичной системы в двоичную.
17. Арифметические основы перевода чисел из двоичной системы в десятичную.
18. Перевод чисел из восьмеричной системы в десятичную и наоборот.
19. Смешанные системы счисления.
20. Арифметические операции в двоичной системе счисления.
21. Форматы представления числовой информации в ЭВМ
22. Арифметические операции. Прямой и обратный коды.
23. Двоичное кодирование символов
24. Основы логики: формы мышления.
25. Алгебра высказываний, логические операции.
26. Логические выражения и таблицы истинности.
27. Логические функции.
28. Логические законы и правила преобразования логических выражений.
29. Базовые логические элементы компьютера.
30. Схемная реализация сумматора и триггера
31. Структура и виды машинных команд
32. Классификация программных средств ЭВМ
33. Системное ПО. Виды современных операционных систем
34. Системы каталогов и файлов
35. Организация доступа к файлам
36. Основные операционные системы. История развития и возможности.
37. Операционная среда Windows. Характеристика и возможности.
38. Служебные программы ОС Windows.
39. Сжатие информации. Назначение и основные программы архивации.
40. Вирусы и основные меры защиты от вирусов.
41. Классификация прикладного программного обеспечения.
42. Табличный процессор. История появления, назначение, интерфейс.
43. Алгоритм. Определение и его свойства.
44. Основные алгоритмические конструкции.
45. Типы данных: простые, структурированные.

46. Устройства объединения ЛВС.
47. Компьютерные сети. Назначение и классификация.
48. Виды топологий. Преимущества и недостатки.
49. Сетевые стандарты. Эталонная модель OSI
50. Интернет: назначение, звенья, адресация.
51. Сетевые протоколы открытых систем.
52. Какие стандартные функции содержатся в файлах `stdio.h` и `conio.h` ?
53. Что означает «инкремент»?
54. Что означает «декремент»?
55. Что означает операция присваивания?
56. Какой спецификатор применяется для ввода и вывода строки символов?
57. Назовите основные операторы циклических процессов.
58. Назовите основные параметры цикла.
59. Как образуется бесконечный цикл и как выйти из него?
60. Какие управляющие средства выполнения программ существуют в языке Си? Объясните их синтаксис.
61. Объясните назначение всех функций, использованных в программах.
62. Какие операции используются в управляющих средствах выполнения программ.
63. Объясните различия между циклами `while` и `do ... while`.
64. Для каких целей используется оператор `switch` ?
65. С какой целью используется индексация массивов?
66. С какой целью используется тип данных?
67. Опишите примененный Вами алгоритм обработки массива.
68. Как определить адрес (i, j) – того элемента в памяти машины?
69. Как распределяется двумерный массив в памяти машины?
70. Что означает транспонирование матрицы?
71. как поменять местами строку и столбец?
72. Какие циклы называются вложенными?
73. Какой переход называется условным / безусловным?
74. Для чего используется оператор `#define`?
75. Назовите операторы условной компиляции.
76. В чем состоит отличие объявления функции от ее определения?
77. Как объявить функцию с переменным числом параметров?
78. Какие типы объектов могут быть использованы в качестве формальных параметров?
79. Что такое локальные объекты? Какова их область видимости и «время жизни»?
80. В чем состоит отличие автоматических переменных от статических?
81. Объясните механизм передачи параметров по значению, по указателю, по ссылке.

Образец билета:

Российский государственный гидрометеорологический университет
 Кафедра информационных технологий и систем безопасности
 ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №1
 Дисциплина «Информатика и программирование»

1. Файловая система компьютера.
2. Методы сжатия информации.

«Утверждаю»

Зав. кафедрой _____ / _____ /

Критерии выставления оценки

Оценка «**отлично**» ставится студенту, ответ которого содержит:

- глубокое знание программного материала, а также основного содержания и новаций лекционного курса по сравнению с учебной литературой;
- знание концептуально-понятийного аппарата всего курса; а также свидетельствует о способности:
- самостоятельно критически оценивать основные положения курса;
- увязывать теорию с практикой.

Оценка «отлично» не ставится в случаях систематических пропусков студентом семинарских и лекционных занятий по неуважительным причинам, а также неправильных ответов на дополнительные вопросы преподавателя.

Оценка «**хорошо**» ставится студенту, ответ которого свидетельствует о полном знании материала по программе, а также содержит в целом правильное, но не всегда точное и аргументированное изложение материала.

Оценка «хорошо» не ставится в случаях пропусков студентом семинарских и лекционных занятий по неуважительным причинам.

Оценка «**удовлетворительно**» ставится студенту, ответ которого содержит:

- поверхностные знания важнейших разделов программы и содержания лекционного курса;
- затруднения с использованием научно-понятийного аппарата и терминологии курса;
- стремление логически четко построить ответ, а также свидетельствует о возможности последующего обучения.

Оценка «**неудовлетворительно**» ставится студенту, имеющему существенные пробелы в знании основного материала по программе, а также допустившему принципиальные ошибки при изложении материала.

Перечень вопросов для промежуточной аттестации экзамен (Разделы 2 и 3, 2 семестр):

1. Основные этапы и лексемы создания программы.
2. Назначение компилятора и компоновщика.
3. Классификация языков программирования.
4. Структурное программирование
5. Переменные, константы, строки.
6. Имена и типы, область видимости.

7. Стандартные математические функции.
8. Стандартные библиотечные функции.
9. Типы данных. Простые типы.
10. Операции над данными, приведение типов.
11. Инкрементные операции языка Си.
12. Арифметические операции языка. Приоритет операций.
13. Бинарные операции.
14. Вывод результатов. Функции вывода.
15. Функции форматированного ввода данных
16. Спецификации полей данных.
17. Функции ввода строк и символов
18. Адресация данных. Операции над адресами.
19. Функции ввода данных.
20. Поразрядные (битовые) операции.
21. Назначение и использование оператора `if`.
22. Вложенные конструкции оператора `if`
23. Оператор для организации циклов `for`.
24. Операторы для организации циклов `while`, `do... while`
25. Операторы передачи управления.
26. Синтаксис использования операторов `break` и `continue`.
27. Директивы условной компиляции.
28. Препроцессоры языка Си.
29. Макросы: назначение и использование
30. Объявления и инициализация массивов.
31. Символьные массивы. Объявления и инициализация.
32. Указатели: назначение, объявления и инициализация. Операции над указателями.
33. Связь массива и указателя. Различие операций `&` и `*`.
34. Типовая структура программы на языке Си с функциями.
35. Функции пользователя: основные определения.
36. Рекурсивные функции.
37. Назначение и использование оператора **`return`**.
38. Передача аргументов в функцию `main()`.
39. Динамическое распределение памяти. Функции для выделения и освобождения памяти.
40. Назначение и использование двойных указателей. Динамические массивы.
41. Структуры языка Си. Инициализация структуры.
42. Массивы структур языка Си.
43. Глобальные и локальные переменные. Правила инициализации.
44. Функции обработки строк `strcmp()`, `strncmp()`.
45. Функции обработки строк `strcpy()`, `strncpy()`.
46. Функции обработки строк `strcat()`, `strncat()`.
47. Назначение и содержание заголовочного файла `ctype.h`
48. Организация работы с файлами: Открытие и закрытие.
49. Организация работы с файлами: режимы работы с файлами.
50. Файловые функции обработки символов и строк.

51. Форматированный ввод/вывод в файл/из файла.
52. Управление экраном, курсором и цветом.
53. Организация видеопамати в текстовом режиме
54. Какой массив называется динамическим?
55. Что такое поток данных?
56. В чем состоит различие ввода данных в стандартном потоке и с внешнего устройства?
57. Напишите команды открытия файла для чтения и записи, закрытия файла.
58. В чем состоит различие команд fputs() и fgets()?
59. Какой список называется линейным?
60. Какой список называется кольцевым?
61. Как проверить: входит ли заданный элемент в список?
62. Как найти список с заданным значением информационного поля?
63. Дайте определение класса и объекта.
64. Дайте определение инкапсуляции и наследования.
65. Какие функции называются перегружаемые и почему?
66. Что общего между классом и структурой в языке C++?
- 67.. Структурные и функциональные схемы ПО.
68. Проектирование структур данных.
69. Проектирование ПО, основанное на декомпозиции данных.
70. Метод пошаговой детализации.
71. Структурные карты Констатайна.
72. Проектирование ПО. Методика Джексона
73. Case- технологии, основанные на методологиях анализа и проектирования.
74. Программирование с защитой от ошибок.
75. Динамические массивы.
76. Динамические структуры.
77. Списки. Линейные и связанные списки.
78. Стеки: Организация хранения в стеке и доступа к элементам.
79. Сортировка данных. Постановка задачи, виды сортировки.
80. Линейная сортировка, метод пузырька.
81. Сортировка вставкой, посредством выбора.
82. Сортировка списков путем слияния.
83. Быстрая сортировка.
84. Челночная сортировка.
85. Алгоритмы поиска.
86. Принципы и технологии программирования в ООП.
87. Методология работы в Visual Studio.
88. Методология объектно-ориентированного программирования.
89. Функции ввода/вывода в C++. Форматы вывода.
90. Понятие класса и объекта в C++.
91. Перегруженные функции.
92. Шаблонные функции.
93. Наследование в C++.
94. Конструкторы и деструкторы в C++.
95. Java: Особенности, загрузка, вывод результатов

96. Java: Пример создания программы.
97. Java: Геттеры и сеттеры. Назначение и примеры.
98. Java: инкапсуляция, полиморфизм, наследование.
99. Java: Обработка исключений.
100. Java: Стандартные пакеты.
101. Паттерны проектирования. AbstractFactory.
102. Паттерны проектирования. Builder.
103. Виды контроля качества разрабатываемого ПО.
104. Понятие структурного тестирования программ
105. Функциональное тестирование программ.
106. Отладка программного обеспечения, виды ошибок.
107. Методы отладки программного обеспечения.
108. Правила составления программного продукта.
109. Инструкция пользователя.

Образец билета:

Российский государственный гидрометеорологический университет
Кафедра информационных технологий и систем безопасности

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №1

Дисциплина «Информатика и программирование»

1. Организация работы с файлами: режимы работы с файлами.
2. Связь массива и указателя. Различие операций & и *.

«Утверждаю»

Зав. кафедрой _____ / _____ /

Критерии выставления оценки

Оценка «отлично» ставится студенту, ответ которого содержит:

- глубокое знание программного материала, а также основного содержания и новаций лекционного курса по сравнению с учебной литературой;
- знание концептуально-понятийного аппарата всего курса; а также свидетельствует о способности:
- самостоятельно критически оценивать основные положения курса;
- увязывать теорию с практикой.

Оценка «отлично» не ставится в случаях систематических пропусков студентом семинарских и лекционных занятий по неуважительным причинам, а также неправильных ответов на дополнительные вопросы преподавателя.

Оценка «хорошо» ставится студенту, ответ которого свидетельствует о полном знании материала по программе, а также содержит в целом правильное, но не всегда точное и аргументированное изложение материала.

Оценка «хорошо» не ставится в случаях пропусков студентом семинарских и лекционных занятий по неуважительным причинам.

Оценка «удовлетворительно» ставится студенту, ответ которого содержит:

- поверхностные знания важнейших разделов программы и содержания лекционного курса;
- затруднения с использованием научно-понятийного аппарата и терминологии курса;
- стремление логически четко построить ответ, а также свидетельствует о возможности последующего обучения.

Оценка «неудовлетворительно» ставится студенту, имеющему существенные пробелы в знании основного материала по программе, а также допустившему принципиальные ошибки при изложении материала.

6. Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Платонов, Ю. М. Информатика [Электронный ресурс] : учебное пособие / Платонов Ю. М., Уткин Ю. Г., Иванов М. И. — М. : Альтаир-МГАВТ, 2014. - Режим оступа: <http://znanium.com/bookread2.php?book=476276>

2. Кувшинов, Д. Р. Основы программирования : учебное пособие для вузов / Д. Р. Кувшинов. — М. : Издательство Юрайт, 2018. — 104 с. — (Серия : Университеты России). — ISBN 978-5-534-07559-5. — Режим доступа : www.biblio-online.ru/book/3D35AAB8-863B-47A8-BA39-ABF5D579204D

3. Малявко, А. А. Формальные языки и компиляторы : учебное пособие для вузов / А. А. Малявко. — М. : Издательство Юрайт, 2018. — 429 с. — (Серия : Университеты России). — ISBN 978-5-534-04288-7. — Режим доступа : www.biblio-online.ru/book/B4D96654-71D5-4748-986D-66E8309C25E3.

4. Огнева, М. В. Программирование на языке с++: практический курс : учебное пособие для бакалавриата и специалитета / М. В. Огнева, Е. В. Кудрина. — М. : Издательство Юрайт, 2018. — 335 с. — (Серия : Бакалавр и специалист). — ISBN 978-5-534-05123-0. — Режим доступа : www.biblio-online.ru/book/7670D7EC-AC37-4675-8EAE-DD671BC6D0E4

5. Малявко, А. А. Формальные языки и компиляторы : учебное пособие для вузов / А. А. Малявко. — М. : Издательство Юрайт, 2018. — 429 с. — (Серия : Университеты России). — ISBN 978-5-534-04288-7. — Режим доступа : www.biblio-online.ru/book/B4D96654-71D5-4748-986D-66E8309C25E3

6. Зыков, С. В. Программирование : учебник и практикум для академического бакалавриата / С. В. Зыков. — М. : Издательство Юрайт, 2018. — 320 с. — (Серия : Бакалавр. Академический курс). — ISBN 978-5-534-02444-9. — Режим доступа : www.biblio-online.ru/book/122D27F3-13E4-4095-8946-C619F0FCC5C3

б) дополнительная литература:

7. Шишкин А.Д., Чернецова Е.А. Практикум "Программирование на языке Си" [Текст] : учеб. пособие / РГГМУ. - Санкт-Петербург : РГГМУ, 2003. - 51 с. - 25.00 р.

8. Шишкин А.Д., Чернецова Е.А. Практикум по дисциплине "Языки программирования". Раздел: Программирование С и С++ [Текст] : практикум / РГГМУ. - 2-е изд., испр. и доп. - Санкт-Петербург : РГГМУ, 2012. - 68 с. - 29.43 р.

9. Шишкин А.Д. Программирование на языке Си [Текст] : учебное пособие / РГГМУ. - Санкт-Петербург : РГГМУ, 2003. - 103 с. - 70.00 р.

10. Назаров, С. В. Программирование в пакетах MS Office [Электронный ресурс] : учеб. пособие / С. В. Назаров, П. П. Мельников, Л. П. Смольников и др.; под ред. С. В. Назарова. - М.: Финансы и статистика, 2007. - 656 с.: ил. - ISBN 978-5-279-02926-6 - Режим доступа: <http://znanium.com/bookread2.php?book=369386>

в) программное обеспечение и Интернет-ресурсы:

Программное обеспечение:

- windows 7
- office 2007
- dr Web
- Dev-C++ GNU General Public License

Интернет-ресурсы

- <https://inf1.info/> - Планета информатики
- <http://taskcode.ru/> - Решение задач по программированию
- <http://moodle.rshu.ru> - система управления курсами РГГМУ

Информационно-справочные системы:

- <https://biblio-online.ru> – ЭБС Юрайт
- <http://znanium.com> – ЭБС Знаниум
- <http://www.prospektnauki.ru> – ЭБС Проспект науки
- <http://elib.rshu.ru> ЭБС ГидроМетеоОнлайн
- <https://нэб.рф> - Национальная электронная библиотека

Профессиональные базы данных

- Профессиональные базы данных не используются

7. Методические указания для обучающихся по освоению дисциплины (модуля)

Виды учебных занятий	Организация деятельности студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометить важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на

	консультации или на лабораторном занятии.
Лабораторные	На лабораторных занятиях выполняются работы по программированию физических и математических задач, изученные во время лекций. Как правило, на каждом занятии студент должен показать результаты выполнения лабораторной преподавателю. После выполнения лабораторной работы студент готовится к ее защите и защищает ее.
Внеаудиторная работа	представляет собой вид занятий, которые каждый студент организует и планирует самостоятельно. Самостоятельная работа студентов включает самостоятельное изучение разделов дисциплины.
Подготовка к экзамену/ зачёту/ курсовой работе	При подготовке к экзамену, зачету и защите курсовой работы необходимо ориентироваться на конспекты лекций, рекомендуемую литературу и выполненные лабораторные работы.

8. Информационные технологии, используемые при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)

Тема (раздел) дисциплины	Образовательные и информационные технологии	Перечень программного обеспечения и информационных справочных систем
Информатика	Лекции. Лабораторные работы.	https://biblio-online.ru http://znanium.com http://www.prospektnauki.ru http://elib.rshu.ru https://нэб.рф windows 7 office 2007 dr Web
Языки программирования	Лекции. Лабораторные работы	https://biblio-online.ru http://znanium.com http://www.prospektnauki.ru http://elib.rshu.ru https://нэб.рф windows 7 office 2007 dr Web Dev-C++ GNU
Методы программирования	Лекции. Лабораторные работы	https://biblio-online.ru http://znanium.com http://www.prospektnauki.ru http://elib.rshu.ru https://нэб.рф windows 7 office 2007 dr Web Dev-C++ GNU

9. Особенности освоения дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Обучение обучающихся с ограниченными возможностями здоровья при необходимости осуществляется на основе адаптированной рабочей программы с использованием специальных методов обучения и дидактических материалов, составленных с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся (обучающегося).

При определении формы проведения занятий с обучающимся-инвалидом учитываются рекомендации, содержащиеся в индивидуальной программе реабилитации инвалида, относительно рекомендованных условий и видов труда. При необходимости для обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья создаются специальные рабочие места с учетом нарушенных функций и ограничений жизнедеятельности.

10. Материально-техническое обеспечение дисциплины

Учебная аудитории для проведения занятий лекционного типа – укомплектована специализированной (учебной) мебелью, набором демонстрационного оборудования и учебно-наглядными пособиями, обеспечивающими тематические иллюстрации, соответствующие рабочим учебным программам дисциплин (модулей).

Учебная аудитории для проведения занятий лабораторного типа - компьютерный класс с ЛВС, связанной Интернетом

Учебная аудитория для групповых и индивидуальных консультаций - укомплектована специализированной (учебной) мебелью, техническими средствами обучения, служащими для представления учебной информации.

Учебная аудитория для текущего контроля и промежуточной аттестации - укомплектована специализированной (учебной) мебелью, техническими средствами обучения, служащими для представления учебной информации.

Помещение для самостоятельной работы – укомплектовано специализированной (учебной) мебелью, оснащено компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечено доступом в электронную информационно-образовательную среду организации

Помещение для хранения и профилактического обслуживания учебного оборудования – укомплектовано специализированной мебелью для хранения оборудования и техническими средствами для его обслуживания.