

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ,
СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ**
**«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»
(СПбГУТ)**

Кафедра _____ Программной инженерии и вычислительной техники
(полное наименование кафедры)



Регистрационный №_24.05/396-Д

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Алгоритмические основы программной инженерии
(наименование дисциплины)

образовательная программа высшего образования

09.03.04 Программная инженерия
(код и наименование направления подготовки / специальности)

бакалавр
(квалификация)

Разработка программного обеспечения и приложений
искусственного интеллекта в киберфизических системах
(направленность / профиль образовательной программы)

очная форма, заочная форма
(форма обучения)

Санкт-Петербург

Рабочая программа дисциплины составлена на основе требований Федерального государственного образовательного стандарта высшего образования по направлению (специальности) подготовки «09.03.04 Программная инженерия», утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 920, и в соответствии с рабочим учебным планом, утвержденным ректором университета.

1. Цели и задачи дисциплины

Целью преподавания дисциплины «Алгоритмические основы программной инженерии» является:

изучение основных понятий и методов теории алгоритмов, используемых в информатике и вычислительной технике; приобретение умений использования их для построения несложных логических моделей предметных областей, реализации логического вывода и оценки вычислительной сложности алгоритмов; получение представление о направлениях развития данной дисциплины и перспективах ее использования в информатике и вычислительной технике.

Эта цель достигается путем решения следующих(ей) задач(и):

приобретения умений использования их для построения несложных логических моделей предметных областей, реализации логического вывода и оценки вычислительной сложности алгоритмов; получение представление о направлениях развития данной дисциплины и перспективах ее использования в информатике и вычислительной технике.

2. Место дисциплины в структуре образовательной программы

Дисциплина «Алгоритмические основы программной инженерии» Б1.О.08 является одной из дисциплин обязательной части учебного плана подготовки бакалавриата по направлению «09.03.04 Программная инженерия». Исходный уровень знаний и умений, которыми должен обладать студент, приступая к изучению данной дисциплины, определяется изучением таких дисциплин, как «Информатика»; «Программирование».

3. Перечень планируемых результатов обучения, соотнесенных с планируемыми результатами освоения образовательной программы

Процесс изучения дисциплины направлен на формирование следующих компетенций:

Таблица 1

№ п/п	Код компетенции	Наименование компетенции
1	ОПК-4	Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью;
2	ОПК-6	Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов
3	ОПК-7	Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой
4	ПК-8	Способность создавать программные интерфейсы
5	УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений

Индикаторы достижения компетенций

Таблица 2

ОПК-4.1	Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы
ОПК-4.2	Умеет применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы
ОПК-4.3	Имеет навыки составления технической документации на различных этапах жизненного цикла информационной системы
ОПК-6.1	Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий
ОПК-6.2	Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ
ОПК-6.3	Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач
ОПК-7.1	Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий
ОПК-7.2	Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ
ОПК-7.3	Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач
ПК-8.1	Знает способы создания программных интерфейсов
ПК-8.2	Умеет создавать интуитивно понятные программные интерфейсы
ПК-8.3	Имеет навыки в создании современных программных интерфейсов
УК-2.1	Знает необходимые для осуществления профессиональной деятельности правовые нормы
УК-2.2	Умеет определять круг задач в рамках избранных видов профессиональной деятельности, планировать собственную деятельность исходя из имеющихся ресурсов; соотносить главное и второстепенное, решать поставленные задачи в рамках избранных видов профессиональной деятельности
УК-2.3	Имеет практический опыт применения нормативной базы и решения задач в области избранных видов профессиональной деятельности

4. Объем дисциплины и виды учебной работы

Очная форма обучения

Таблица 3

Вид учебной работы		Всего часов	Семестры
			2
Общая трудоемкость	4 ЗЕТ	144	144
Контактная работа с обучающимися		68.35	68.35
в том числе:			
Лекции		26	26
Практические занятия (ПЗ)		22	22
Лабораторные работы (ЛР)		18	18
Защита контрольной работы			-
Защита курсовой работы			-
Защита курсового проекта			-

Промежуточная аттестация	2.35	2.35
Самостоятельная работа обучающихся (СРС)	42	42
в том числе:		
Курсовая работа		-
Курсовой проект		-
И / или другие виды самостоятельной работы: подготовка к лабораторным работам, практическим занятиям, контрольным работам, изучение теоретического материала	42	42
Подготовка к промежуточной аттестации	33.65	33.65
Вид промежуточной аттестации		Экзамен

Заочная форма обучения

Таблица 4

Вид учебной работы		Всего часов	Семестры		
			усЗ	3	4
Общая трудоемкость	4 ЗЕТ	144	8	64	72
Контактная работа с обучающимися		14.65	8	4	2.65
в том числе:					
Лекции		4	4	-	-
Практические занятия (ПЗ)		4	-	4	-
Лабораторные работы (ЛР)		4	4	-	-
Защита контрольной работы		0.3	-	-	0.3
Защита курсовой работы			-	-	-
Защита курсового проекта			-	-	-
Промежуточная аттестация		2.35	-	-	2.35
Самостоятельная работа обучающихся (СРС)		120.35	-	60	60.35
в том числе:					
Курсовая работа			-	-	-
Курсовой проект			-	-	-
И / или другие виды самостоятельной работы: подготовка к лабораторным работам, практическим занятиям, контрольным работам, изучение теоретического материала		120.35	-	60	60.35
Подготовка к промежуточной аттестации		9	-	-	9
Вид промежуточной аттестации			-	-	Экзамен

5. Содержание дисциплины

5.1. Содержание разделов дисциплины.

Таблица 5

№ п/п	Наименование раздела дисциплины	Содержание раздела	№ семестра		
			очная	очно-заочная	заочная
1	Раздел 1. Введение в программную инженерию.	Общее введение в программную инженерию. Разработке программного обеспечения на основе инженерных принципов. История развития средств вычислительной техники. Исторические аспекты теории алгоритмов.	2		3

2	Раздел 2. Программное обеспечение ЭВМ и автоматизированных систем управления (ПО)	Введение основных понятий - программа, программное обеспечение, программные средства, программный комплекс. Инженерный подход к конструированию ПО. Классификация ПО по назначению. Системное ПО - классификация, использование. Операционные системы (ОС) - их основные функции, примеры. Прикладное ПО - классификация, основные функции, примеры. Инструментальное ПО. Классификация языков программирования. Примеры. Этапы подготовки исполняемого кода. Назначение транслятора, компоновщика, отладчика. Интегрированные среды разработки программ, их состав. Автоматическая генерация кода программы. CASE-средства. Использование языка UML для генерации кода.	2		3
3	Раздел 3. Основы теории алгоритмов	Алгоритм. Виды алгоритмов и формы их представления. Основные характеристики и свойства алгоритмов. Главные объекты исследования - информация и алгоритмы ее обработки. Определение алгоритма по Колмогорову и Маркову. История вопроса. Применение алгоритма Евклида о нахождении наибольшего общего делителя двух чисел и великой теоремы Ферма в современных алгоритмах. Постановка основных проблем, стоящих перед теоретической информатикой с точки зрения компьютерных наук: 1. Алгоритмически неразрешимые задачи. Теорема Геделя о неполноте. Примеры алгоритмически неразрешимых задач. Проблема останова машины Тьюринга. Вычисление совершенных чисел. Десятая проблема Гильберта. Проблема соответствия Поста над алфавитом . 2. Использование теории вычислимости для определения алгоритмически неразрешимых проблем. 3. Использование теории сложности для определения сложностного класса алгоритмически разрешимой задачи. 4. Особенности задач, занимающихся случайными процессами	2		3
4	Раздел 4. Различные математические формализмы, положенные в основу вычислительных процессов, используемых в программировании.	Рассматриваются основные парадигмы программирования. Проводится их сравнительный анализ. Даются методы оценки каждой вычислительной модели.	2		3

5	Раздел 5. Современные алгоритмы и теория связи.	Особенности работы с компьютерными сетями. Распределенные вычисления. Технологии разработки программного обеспечения и оптимизация алгоритмов. Примеры современных алгоритмических идей и их связь с теорией простых чисел. Примеры. Криптосистема RSA. Использование в ней малой теоремы Ферма и алгоритма Евклида. Вероятностный тест Миллера-Рабина для криптографических алгоритмов. Алгоритм Рабина- Карпа. Генетические алгоритмы. Муравьиные алгоритмы. Новые технологии – новые проблемы в области разработки и оптимизации алгоритмов.	2		3
6	Раздел 6. Множества.	Определение. Теоретико-множественные операции и их свойства. Примеры и упражнения. Элементы комбинаторики в приложении к множествам. Примеры и упражнения.	2		3
7	Раздел 7. Формальные теории. Формальные системы и формальные языки	Общая схема построения формальной системы. Язык системы. Аксиомы системы. Правила вывода. Формальный язык информатики и способы представления объектов, задач, целей. Алфавиты, слова, языки. Определения. Примеры алфавитов, используемых в программировании на компьютере. Упражнения. Алгоритмические проблемы. Проблема принадлежности. Проблема выполнимости. Оптимизационная проблема. Примеры. Упражнения. Сложность по Колмогорову. Ее применение для измерения объема информации и уровня случайности слова. Примеры. Упражнения.	2		3
8	Раздел 8. Конечные автоматы как модель простых вычислений.	Использование конечных автоматов для распознавания языков.	2		3
9	Раздел 9. Машины Тьюринга	Структура машины Тьюринга. Такт работы машины Тьюринга. Пример выполнения программы. Упражнения	2		3
10	Раздел 10. Теория вычислимости	Метод диагонализации. Метод сводимости. Их использование	2		3
11	Раздел 11. Теория сложности	Классы сложности P, NPC и NP. Их сравнительный анализ. Их связь с ресурсами компьютера. Сложность по времени. Сложность по памяти. Основная вычислительная модель абстрактной теории сложности - многоленточная машина Тьюрин-га.	2		3
12	Раздел 12. Компьютерные сети. Вопросы сетевой компьютерной безопасности	Основные понятия безопасности. Системный подход к обеспечению безопасности. Базовые технологии безопасности. Алгоритмы шифрования. Защита информации.	2		3

5.2. Междисциплинарные связи с обеспечиваемыми (последующими) дисциплинами.

Таблица 6

№ п/п	Наименование обеспечиваемых (последующих) дисциплин
-------	---

1	Алгоритмы и структуры данных
2	Математические методы и алгоритмы функционирования киберфизических систем

5.3. Разделы дисциплин и виды занятий.

Очная форма обучения

Таблица 7

№ п/п	Наименование раздела дисциплины	Лек-ции	Практ. занятия	Лаб. занятия	Семи-нары	СРС	Всего часов
1	Раздел 1. Введение в программную инженерию.	2				2	4
2	Раздел 2. Программное обеспечение ЭВМ и автоматизированных систем управления (ПО)	2	8	2		6	18
3	Раздел 3. Основы теории алгоритмов	4		4		2	10
4	Раздел 4. Различные математические формализмы, положенные в основу вы- числительных процессов, используемых в программировании.	2		2		2	6
5	Раздел 5. Современные алгоритмы и теория связи.	2	4	2		2	10
6	Раздел 6. Множества.	2				2	4
7	Раздел 7. Формальные теории. Формальные системы и формальные языки	2	4			10	16
8	Раздел 8. Конечные автоматы как модель простых вычислений.	2	4	4		2	12
9	Раздел 9. Машины Тьюринга	2				8	10
10	Раздел 10. Теория вычислимости	2		2		2	6
11	Раздел 11. Теория сложности	2	2	2		2	8
12	Раздел 12. Компьютерные сети. Вопросы сетевой компьютерной безопасности	2				2	4
Итого:		26	22	18	-	42	108

Заочная форма обучения

Таблица 8

№ п/п	Наименование раздела дисциплины	Лек-ции	Практ. занятия	Лаб. занятия	Семи-нары	СРС	Всего часов
1	Раздел 1. Введение в программную инженерию.	0.3				4	4.3
2	Раздел 2. Программное обеспечение ЭВМ и автоматизированных систем управления (ПО)	0.3	1.5	0.5		12	14.3

3	Раздел 3. Основы теории алгоритмов	0.7		1		6	7.7
4	Раздел 4. Различные математические формализмы, положенные в основу вычислительных процессов, используемых в программировании.	0.3		0.5		6	6.8
5	Раздел 5. Современные алгоритмы и теория связи.	0.3	0.5	0.5		6	7.3
6	Раздел 6. Множества.	0.3				6	6.3
7	Раздел 7. Формальные теории. Формальные системы и формальные языки	0.3	1			20	21.3
8	Раздел 8. Конечные автоматы как модель простых вычислений.	0.3	0.5	0.5		8	9.3
9	Раздел 9. Машины Тьюринга	0.3				30	30.3
10	Раздел 10. Теория вычислимости	0.3		0.5		8	8.8
11	Раздел 11. Теория сложности	0.3	0.5	0.5		6.35	7.65
12	Раздел 12. Компьютерные сети. Вопросы сетевой компьютерной безопасности	0.3				8	8.3
Итого:		4	4	4	-	120.35	132.35

6. Лекции

Очная форма обучения

Таблица 9

№ п/п	Номер раздела	Тема лекции	Всего часов
1	1	Общее введение в программную инженерию. Разработке программного обеспечения на основе инженерных принципов. История развития средств вычислительной техники. Исторические аспекты теории алгоритмов.	2
2	2	Введение основных понятий - программа, программное обеспечение, программные средства, программный комплекс. Инженерный подход к конструированию ПО. Классификация ПО по назначению. Системное ПО - классификация, использование. Операционные системы (ОС) - их основные функции, примеры. Прикладное ПО - классификация, основные функции, примеры. Инструментальное ПО. Классификация языков программирования. Примеры. Этапы подготовки исполняемого кода. Назначение транслятора, компоновщика, отладчика. Интегрированные среды разработки программ, их состав. Автоматическая генерация кода программы. CASE-средства. Использование языка UML для генерации кода.	2

3	3	Алгоритм. Виды алгоритмов и формы их представления. Основные характеристики и свойства алгоритмов. Главные объекты исследования - информация и алгоритмы ее обработки. Определение алгоритма по Колмогорову и Маркову. История вопроса. Применение алгоритма Евклида о нахождении наибольшего общего делителя двух чисел и великой теоремы Ферма в современных алгоритмах. Постановка основных проблем, стоящих перед теоретической информатикой с точки зрения компьютерных наук: 1. Алгоритмически неразрешимые задачи. Теорема Геделя о неполноте. Примеры алгоритмически неразрешимых задач. Проблема останова машины Тьюринга. Вычисление совершенных чисел. Десятая проблема Гильберта. Проблема соответствия Поста над алфавитом . 2. Использование теории вычислимости для определения алгоритмически неразрешимых проблем.	2
4	3	3. Использование теории сложности для определения сложностного класса алгоритмически разрешимой задачи. 4. Особенности задач, занимающихся случайными процессами	2
5	4	Рассматриваются основные парадигмы программирования. Проводится их сравнительный анализ. Даются методы оценки каждой вычислительной модели.	2
6	5	Особенности работы с компьютерными сетями. Распределенные вычисления. Технологии разработки программного обеспечения и оптимизация алгоритмов. Примеры современных алгоритмических идей и их связь с теорией простых чисел. Примеры. Криптосистема RSA. Использование в ней малой теоремы Ферма и алгоритма Евклида. Вероятностный тест Миллера Рабина для криптографических алгоритмов. Алгоритм Рабина- Карпа. Генетические алгоритмы. Муравьиные алгоритмы. Новые технологии - новые проблемы в области разработки и оптимизации алгоритмов.	2
7	6	Определение. Теоретико-множественные операции и их свойства. Примеры и упражнения. Элементы комбинаторики в приложении к множествам. Примеры и упражнения.	2
8	7	Общая схема построения формальной системы. Язык системы. Аксиомы системы. Правила вывода. Формальный язык информатики и способы представления объектов, задач, целей. Алфавиты, слова, языки. Определения. Примеры алфавитов, используемых в программировании на компьютере. Упражнения. Алгоритмические проблемы. Проблема принадлежности. Проблема выполнимости. Оптимизационная проблема. Примеры. Упражнения. Сложность по Колмогорову. Ее применение для измерения объема информации и уровня случайности слова. Примеры. Упражнения.	2
9	8	Использование конечных автоматов для распознавания языков.	2
10	9	Структура машины Тьюринга. Такт работы машины Тьюринга. Пример выполнения программы. Упражнения	2
11	10	Метод диагонализации. Метод сводимости. Их использование	2
12	11	Классы сложности P, NPC и NP. Их сравнительный анализ. Их связь с ресурсами компьютера. Сложность по времени. Сложность по памяти. Основная вычислительная модель абстрактной теории сложности - многоленточная машина Тьюринга.	2
13	12	Основные понятия безопасности. Системный подход к обеспечению безопасности. Базовые технологии безопасности. Алгоритмы шифрования. Защита информации.	2
Итого:			26

№ п/п	Номер раздела	Тема лекции	Всего часов
1	1	Общее введение в программную инженерию. Разработке программного обеспечения на основе инженерных принципов. История развития средств вычислительной техники. Исторические аспекты теории алгоритмов.	0.3
2	2	Введение основных понятий - программа, программное обеспечение, программные средства, программный комплекс. Инженерный подход к конструированию ПО. Классификация ПО по назначению. Системное ПО - классификация, использование. Операционные системы (ОС) - их основные функции, примеры. Прикладное ПО - классификация, основные функции, примеры. Инструментальное ПО. Классификация языков программирования. Примеры. Этапы подготовки исполняемого кода. Назначение транслятора, компоновщика, отладчика. Интегрированные среды разработки программ, их состав. Автоматическая генерация кода программы. CASE-средства. Использование языка UML для генерации кода.	0.3
3	3	Алгоритм. Виды алгоритмов и формы их представления. Основные характеристики и свойства алгоритмов. Главные объекты исследования - информация и алгоритмы ее обработки. Определение алгоритма по Колмогорову и Маркову. История вопроса. Применение алгоритма Евклида о нахождении наибольшего общего делителя двух чисел и великой теоремы Ферма в современных алгоритмах. Постановка основных проблем, стоящих перед теоретической информатикой с точки зрения компьютерных наук: 1. Алгоритмически неразрешимые задачи. Теорема Геделя о неполноте. Примеры алгоритмически неразрешимых задач. Проблема останова машины Тьюринга. Вычисление совершенных чисел. Десятая проблема Гильберта. Проблема соответствия Поста над алфавитом . 2. Использование теории вычислимости для определения алгоритмически неразрешимых проблем.	0.4
4	3	3. Использование теории сложности для определения сложностного класса алгоритмически разрешимой задачи. 4. Особенности задач, занимающихся случайными процессами	0.3
5	4	Рассматриваются основные парадигмы программирования. Проводится их сравнительный анализ. Даются методы оценки каждой вычислительной модели.	0.3
6	5	Особенности работы с компьютерными сетями. Распределенные вычисления. Технологии разработки программного обеспечения и оптимизация алгоритмов. Примеры современных алгоритмических идей и их связь с теорией простых чисел. Примеры. Криптосистема RSA. Использование в ней малой теоремы Ферма и алгоритма Евклида. Вероятностный тест Миллера Рабина для криптографических алгоритмов. Алгоритм Рабина- Карпа. Генетические алгоритмы. Муравьиные алгоритмы. Новые технологии - новые проблемы в области разработки и оптимизации алгоритмов.	0.3
7	6	Определение. Теоретико-множественные операции и их свойства. Примеры и упражнения. Элементы комбинаторики в приложении к множествам. Примеры и упражнения.	0.3

8	7	Общая схема построения формальной системы. Язык системы. Аксиомы системы. Правила вывода. Формальный язык информатики и способы представления объектов, задач, целей. Алфавиты, слова, языки. Определения. Примеры алфавитов, используемых в программировании на компьютере. Упражнения. Алгоритмические проблемы. Проблема принадлежности. Проблема выполнимости. Оптимизационная проблема. Примеры. Упражнения. Сложность по Колмогорову. Ее применение для измерения объема информации и уровня случайности слова. Примеры. Упражнения.	0.3
9	8	Использование конечных автоматов для распознавания языков.	0.3
10	9	Структура машины Тьюринга. Такт работы машины Тьюринга. Пример выполнения программы. Упражнения	0.3
11	10	Метод диагонализации. Метод сводимости. Их использование	0.3
12	11	Классы сложности P, NPC и NP. Их сравнительный анализ. Их связь с ресурсами компьютера. Сложность по времени. Сложность по памяти. Основная вычислительная модель абстрактной теории сложности - многоленточная машина Тьюринга.	0.3
13	12	Основные понятия безопасности. Системный подход к обеспечению безопасности. Базовые технологии безопасности. Алгоритмы шифрования. Защита информации.	0.3
Итого:			4

7. Лабораторный практикум

Очная форма обучения

Таблица 11

№ п/п	Номер раздела	Наименование лабораторной работы	Всего часов
1	2	Знакомство и работа с операционной системой Linux. Работа в терминальном режиме. Команды ОС Linux. Создание скриптов. Работа с математической системой Maxima. Простые вычисления. Решение уравнений. Интегрирование и дифференцирование. Векторы и матрицы. Построение графиков. Работа с файлами. Контрольный тест Работа с текстовым процессором Word. Создание, хранение, редактирование и управление документами. Использование шаблонов и мастеров. Форматирование документов. (Работа с колонками, выравнивание, нумерация, создание колонтитулов) Создание и редактирование таблиц и их модификация. Использование стандартных формул для расчетов данных. Построение диаграмм. Создание маркированных и нумерованных списков. Создание форм и диалоговых окон. Создание указателей и оглавлений. Контрольный тест Работа с табличным процессором Excel. Ввод и редактирование данных. Управление отображением рабочего листа. Работа с многими окнами. Форматирование рабочего листа. Работа с формулами. Использование функций.	2
2	3	Основы теории алгоритмов	4
3	4	Различные математические формализмы, положенные в основу вычислительных процессов, используемых в программировании.	2
4	5	Современные алгоритмы и теория связи	2
5	8	Конечные автоматы как модель простых вычислений.	4
6	10	Теория вычислимости	2
7	11	Теория сложности	2

Итого:	18
--------	----

Заочная форма обучения

Таблица 12

№ п/п	Номер раздела	Наименование лабораторной работы	Всего часов
1	2	Знакомство и работа с операционной системой Linux. Работа в терминальном режиме. Команды ОС Linux. Создание скриптов. Работа с математической системой Maxima. Простые вычисления. Решение уравнений. Интегрирование и дифференцирование. Векторы и матрицы. Построение графиков. Работа с файлами. Контрольный тест Работа с текстовым процессором Word. Создание, хранение, редактирование и управление документами. Использование шаблонов и мастеров. Форматирование документов. (Работа с колонками, выравнивание, нумерация, создание колонтитулов) Создание и редактирование таблиц и их модификация. Использование стандартных формул для расчетов данных. Построение диаграмм. Создание маркированных и нумерованных списков. Создание форм и диалоговых окон. Создание указателей и оглавлений. Контрольный тест Работа с табличным процессором Excel. Ввод и редактирование данных. Управление отображением рабочего листа. Работа с многими окнами. Форматирование рабочего листа. Работа с формулами. Использование функций.	0.5
2	3	Основы теории алгоритмов	1
3	4	Различные математические формализмы, положенные в основу вычислительных процессов, используемых в программировании.	0.5
4	5	Современные алгоритмы и теория связи	0.5
5	8	Конечные автоматы как модель простых вычислений.	0.5
6	10	Теория вычислимости	0.5
7	11	Теория сложности	0.5
Итого:			4

8. Практические занятия (семинары)

Очная форма обучения

Таблица 13

№ п/п	Номер раздела	Тема занятия	Всего часов
1	2	Работа с математической системой Maxima. Простые вычисления. Решение уравнений. Интегрирование и дифференцирование. Векторы и матрицы. Построение графиков. Работа с файлами. Контрольный тест	2
2	2	Работа с табличным процессором . Ввод и редактирование данных. Управление отображением рабочего листа. Работа с многими окнами. Форматирование рабочего листа. Работа с формулами. Использование функций. Создание диаграмм. Форматирование диаграмм. Размещение графических объектов. Структуризация рабочих листов. Автоматическое подведение промежуточных итогов. Работа с анализом рабочего листа. Создание элементов управления рабочим листом. Возможности сводных таблиц. Управление списками и базами данных. Сортировка, поиск, фильтрация данных. Контрольный тест	2

3	2	Знакомство и работа с операционной системой Linux. Работа в терминальном режиме. Команды ОС Linux. Создание скриптов.	2
4	2	Работа с текстовым процессором. Создание, хранение, редактирование и управление документами. Использование шаблонов и мастеров. Форматирование документов.(Работа с колонками, выравнивание, нумерация, создание колонтитулов). Создание и редактирование таблиц и их модификация. Использование стандартных формул для расчетов данных. Построение диаграмм. Создание маркированных и нумерованных списков. Создание форм и диалоговых окон. Создание указателей и оглавлений. Контрольный тест	2
5	5	Рандомизированный коммуникационный протокол связи	2
6	5	Цифровая подпись	2
7	7	Алгоритмические проблемы. Работа с графами. Работа с алфавитами	2
8	7	Сложность по Колмогорову. Определение коэффициента сжатия. Сопоставление различных методов сжатия.	2
9	8	Моделирование процессов вычислений с помощью конечных автоматов	2
10	8	Разработка машины Тьюринга для некоторого заданного языка	2
11	11	Теория сложности. Работа с классами сложности	2
Итого:			22

Заочная форма обучения

Таблица 14

№ п/п	Номер раздела	Тема занятия	Всего часов
1	2	Работа с математической системой Maxima. Простые вычисления. Решение уравнений. Интегрирование и дифференцирование. Векторы и матрицы. Построение графиков. Работа с файлами. Контрольный тест	0.5
2	2	Знакомство и работа с операционной системой Linux. Работа в терминальном режиме. Команды ОС Linux. Создание скриптов.	0.5
3	2	Работа с текстовым процессором. Создание, хранение, редактирование и управление документами. Использование шаблонов и мастеров. Форматирование документов.(Работа с колонками, выравнивание, нумерация, создание колонтитулов). Создание и редактирование таблиц и их модификация. Использование стандартных формул для расчетов данных. Построение диаграмм. Создание маркированных и нумерованных списков. Создание форм и диалоговых окон. Создание указателей и оглавлений. Контрольный тест	0.5
4	5	Цифровая подпись	0.5
5	7	Алгоритмические проблемы. Работа с графами	0.5
6	7	Работа с алфавитами	0.5
7	8	Моделирование процессов вычислений с помощью конечных автоматов	0.5
8	11	Теория сложности. Работа с классами сложности	0.5
Итого:			4

9. Примерная тематика курсовых проектов (работ)

Рабочим учебным планом не предусмотрено

10. Самостоятельная работа

Очная форма обучения

Таблица 15

№ п/п	Номер раздела	Содержание самостоятельной работы	Форма контроля	Всего часов
1	1	Введение в программную инженерию	Опрос	2
2	2	Изучение ОС Linux	Защита лабораторных работ	2
3	2	Изучение текстового процессора и табличного процессора	Защита лабораторных работ, контроль-ный тест	2
4	2	Изучение математического пакета wxmaxima.	Защита лабораторных работ, контроль-ный тест	2
5	3	Основы теории алгоритмов	Опрос	2
6	4	Различные математические формализмы, положенные в основу вычислительных процессов, используемых в программировании	Опрос	2
7	5	Современные алгоритмы и теория связи	Опрос	2
8	6	Множества	Опрос	2
9	7	Провести сравнение алгоритмов, используя классификацию скоростей роста сложности алгоритмов для некоторых заданных функций.	Доклад на семинаре, промежуточный зачет	2
10	7	Работа с алфавитами. Подсчет количества слов определенной длины над заданными алфавитами.	Защита реферата	2
11	7	Работа с алфавитами. Разработка представления графа над заданным алфавитом.	Защита работы на семинаре	2
12	7	Работа с алфавитами. Решение задач типа: Нахождение максимально возможного числа подслов в некотором слове для заданного алфавита	Защита реферата	2
13	7	Работа с языками. Решение задач типа: Даны языки L1 и L2. Определить, какие слова принадлежат языку L1L2?	Защита работы на семинаре	2
14	8	Конечные автоматы как модель простых вычислений	Опрос	2
15	9	Машина Тьюринга. Анализ символов. Задачи типа: $A = \{a, b, c\}$ Перенести первый символ непустого слова P в его конец.	Защита работы	2
16	9	Машина Тьюринга. Сравнение символов. Стирание слова. Задачи типа: $A = \{a, b, c\}$. Если первый и последний символы непустого слова P одинаковы, то слово изменить не надо. Иначе заменить его на пустое слово.	Защита работы. Доклад на семинаре	2
17	9	Машина Тьюринга. Удаление символа из слова. Задачи типа: $F = \{a, b\}$. Удалить из слова P его второй символ, если такой есть.	Защита работы. Доклад на семинаре	2
18	9	Машина Тьюринга. Вставка символа в слово. Задачи типа: $A = \{a, b, c\}$. Если P непустое слово, то за его первым символом вставить символ a.	Защита реферата	2

19	10	Теория вычислимости	Опрос	2
20	11	Сложность по Колмогорову. Решение задач типа: Найти два слова, такие, что метод сжатия повторениями под слова дает существенно более короткое представление слова x , чем метод сжатия разложением на множители.	Доклад на семинаре, промежуточный зачет	2
21	12	Компьютерные сети. Вопросы сетевой компьютерной безопасности	Опрос	2
Итого:				42

Заочная форма обучения

Таблица 16

№ п/п	Номер раздела	Содержание самостоятельной работы	Форма контроля	Всего часов
1	1	Введение в программную инженерию	Опрос	4
2	2	Изучение ОС Linux	Защита лабораторных работ	4
3	2	Изучение текстового процессора и табличного процессора	Защита лабораторных работ, контроль-ный тест	4
4	2	Изучение математического пакета wxmaxima.	Защита лабораторных работ, контроль-ный тест	4
5	3	Основы теории алгоритмов	Опрос	6
6	4	Различные математические формализмы, положенные в основу вычислительных процессов, используемых в программировании	Опрос	6
7	5	Современные алгоритмы и теория связи	Опрос	6
8	6	Множества	Опрос	6
9	7	Провести сравнение алгоритмов, используя классификацию скоростей роста сложности алгоритмов для некоторых заданных функций.	Доклад на семинаре, промежуточный зачет	4
10	7	Работа с алфавитами. Подсчет количества слов определенной длины над заданными алфавитами.	Защита реферата	4
11	7	Работа с алфавитами. Разработка представления графа над заданным алфавитом.	Защита работы на семинаре	4
12	7	Работа с алфавитами. Решение задач типа: Нахождение максимально возможного числа подслов в некотором слове для заданного алфавита	Защита реферата	4
13	7	Работа с языками. Решение задач типа: Даны языки L_1 и L_2 . Определить, какие слова принадлежат языку L_1L_2 ?	Защита работы на семинаре	4
14	8	Конечные автоматы как модель простых вычислений	Опрос	8
15	9	Машина Тьюринга. Анализ символов. Задачи типа: $A = \{a, b, c\}$ Перенести первый символ непустого слова P в его конец.	Защита работы	8

16	9	Машина Тьюринга. Сравнение символов. Стирание слова. Задачи типа: $A=\{a,b,c\}$. Если первый и последний символы непустого слова P одинаковы, то слово изменить не надо. Иначе заменить его на пустое слово.	Защита работы. Доклад на семинаре	6
17	9	Машина Тьюринга. Удаление символа из слова. Задачи типа: $F=\{a,b\}$. Удалить из слова P его второй символ, если такой есть.	Защита работы. Доклад на семинаре	6
18	9	Машина Тьюринга. Вставка символа в слово. Задачи типа: $A=\{a,b,c\}$. Если P непустое слово, то за его первым символом вставить символ a .	Защита реферата	6
19	9	Машина Тьюринга. Анализ символов. Задачи типа: $A=\{a,b,c\}$ Перенести первый символ непустого слова P в его конец.	Опрос	4
20	10	Теория вычислимости	Опрос	8
21	11	Сложность по Колмогорову. Решение задач типа: Найти два слова, такие, что метод сжатия повторениями под слова дает существенно более короткое представление слова x , чем метод сжатия разложением на множители.	Доклад на семинаре, промежуточный зачет	6.35
22	12	Компьютерные сети. Вопросы сетевой компьютерной безопасности	Опрос	8
Итого:				120.35

11. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Для самостоятельной работы по дисциплине рекомендовано следующее учебно-методическое обеспечение:

- Положение о самостоятельной работе студентов в Санкт-Петербургском государственном университете телекоммуникаций им. проф. М.А. Бонч-Бруевича;
- рекомендованная основная и дополнительная литература;
- конспект занятий по дисциплине;
- слайды-презентации и другой методический материал, используемый на занятиях;
- методические рекомендации по подготовке письменных работ, требования к их содержанию и оформлению (реферат, эссе, контрольная работа) ;
- фонды оценочных средств;
- методические указания к выполнению лабораторных работ для студентов;

12. Фонд оценочных средств для проведения промежуточной аттестации обучающихся

Фонд оценочных средств разрабатывается в соответствии с Методическими рекомендациями по формированию ФОС и приказом Минобрнауки России от 5 апреля 2017г. № 301 г. Москва "Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры" и является приложением к рабочей программе дисциплины.

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине включает в себя:

- перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы;
- описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
- типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы;
- методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Для каждого результата обучения по дисциплине определяются показатели и критерии оценки сформированности компетенций на различных этапах их формирования, шкалы и процедуры оценивания.

13. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины

13.1. Основная литература:

1. Новиков, А.

Дискретная математика : [Электронный ресурс] : учебник для вузов. 2-е изд. / А. Новиков. - СПб. : Питер, 2021. - 432 с. - URL: <http://ibooks.ru/reading.php?productid=377364>. - ISBN 978-5-4461-9746-0 : Б. ц. Стандарт третьего поколения.

2. Каймин, В. А.

Информатика : [Электронный ресурс] : учебник / В. А. Каймин. - 6-е изд. - М. : Инфра-М, 2016. - 285 с. - URL: <http://ibooks.ru/reading.php?productid=361264>. - ISBN 978-5-16-003778-3 : Б. ц.

13.2. Дополнительная литература:

1. Ланских, В. Г.

Основы теории алгоритмов : [Электронный ресурс] : учебное пособие / В. Г. Ланских. - Киров : ВятГУ, 2017. - 78 с. - URL: <https://e.lanbook.com/book/164446>. - Б. ц. Книга из коллекции ВятГУ - Информатика

14. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

Ресурсы информационно-телекоммуникационной сети интернет из указанного перечня являются рекомендуемыми дополнительными (вспомогательными) источниками официальной информации, размещенной на легальных основаниях с открытым доступом. За полноту содержания и качество работы сайтов несет ответственность правообладатель.

Наименование ресурса	Адрес
Официальный сайт СПбГУТ	sut.ru/
Сайт кафедры ПИВТ	pivt.sut.ru/

15. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем.

15.1. Программное обеспечение дисциплины:

- Code::Blocks
- Evince
- Libre Office
- Linux Debian
- Maxima
- Netbeans
- Компилятор gcc

15.2. Информационно-справочные системы:

- ЭБС iBooks (<https://ibooks.ru>)
- ЭБС Лань (<https://e.lanbook.com/>)
- ЭБС СПбГУТ (<http://lib.spbgut.ru>)

16. Методические указания для обучающихся по освоению дисциплины

15.1. Планирование и организация времени, необходимого для изучения дисциплины

Важным условием успешного освоения дисциплины «Алгоритмические основы программной инженерии» является создание системы правильной организации труда, позволяющей распределить учебную нагрузку равномерно в соответствии с графиком образовательного процесса. Большую помощь в этом может оказать составление плана работы на семестр, месяц, неделю, день. Его наличие позволит подчинить свободное время целям учебы, трудиться более успешно и эффективно. Нужно осуществлять самоконтроль, который является необходимым условием успешной учебы. Все задания к практическим занятиям, а также задания, вынесенные на самостоятельную работу, рекомендуется выполнять непосредственно после соответствующей темы лекционного курса, что способствует лучшему усвоению материала, позволяет своевременно выявить и устранить «пробелы» в знаниях, систематизировать ранее пройденный материал, на его основе приступить к овладению новыми знаниями и навыками.

Система университетского обучения основывается на рациональном сочетании нескольких видов учебных занятий (в первую очередь, лекций и практических занятий), работа на которых обладает определенной спецификой.

15.2. Подготовка к практическим занятиям

Подготовку к каждому практическому занятию студент должен начать с ознакомления с планом практического занятия, который отражает содержание предложенной темы. Тщательное продумывание и изучение вопросов плана основывается на проработке текущего материала лекции, а затем изучения обязательной и дополнительной литературы, рекомендованной к данной теме. Все новые понятия по изучаемой теме необходимо выучить наизусть и внести в глоссарий, который целесообразно вести с самого начала изучения курса.

Результат такой работы должен проявиться в способности студента свободно ответить на теоретические вопросы практикума, его выступлении и участии в коллективном обсуждении вопросов изучаемой темы, правильном выполнении практических заданий и контрольных работ.

В процессе подготовки к практическим занятиям, студентам необходимо обратить особое внимание на самостоятельное изучение рекомендованной литературы. При всей полноте конспектирования лекции в ней невозможно изложить весь материал из-за лимита аудиторных часов. Поэтому самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами периодических изданий и Интернета является наиболее эффективным методом получения дополнительных знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у студентов свое отношение к конкретной проблеме.

15.3. Рекомендации по работе с литературой

Работу с литературой целесообразно начать с изучения общих работ по теме, а также учебников и учебных пособий. Далее рекомендуется перейти к анализу монографий и статей, рассматривающих отдельные аспекты проблем, изучаемых в рамках курса, а также официальных материалов и неопубликованных документов (научно-исследовательские работы, диссертации), в которых могут содержаться основные вопросы изучаемой проблемы.

Работу с источниками надо начинать с ознакомительного чтения, т.е. просмотреть текст, выделяя его структурные единицы. При ознакомительном чтении закладками отмечаются те страницы, которые требуют более внимательного изучения. В зависимости от результатов ознакомительного чтения выбирается дальнейший способ работы с источником. Если для разрешения поставленной задачи требуется изучение некоторых фрагментов текста, то используется метод выборочного чтения. Если в книге нет подробного оглавления, следует обратить внимание ученика на предметные и именные указатели.

Избранные фрагменты или весь текст (если он целиком имеет отношение к теме) требуют вдумчивого, неторопливого чтения с «мысленной проработкой» материала. Такое чтение предполагает выделение: 1) главного в тексте; 2) основных аргументов; 3) выводов. Особое внимание следует обратить на то, вытекает тезис из аргументов или нет. Необходимо также проанализировать, какие из утверждений автора носят проблематичный, гипотетический характер и уловить скрытые вопросы.

Понятно, что умение таким образом работать с текстом приходит далеко не

сразу. Наилучший способ научиться выделять главное в тексте, улавливать проблематичный характер утверждений, давать оценку авторской позиции – это сравнительное чтение, в ходе которого студент знакомится с различными мнениями по одному и тому же вопросу, сравнивает весомость и доказательность аргументов сторон и делает вывод о наибольшей убедительности той или иной позиции.

Если в литературе встречаются разные точки зрения по тому или иному вопросу из-за сложности прошедших событий и правовых явлений, нельзя их отвергать, не разобравшись. При наличии расхождений между авторами необходимо найти рациональное зерно у каждого из них, что позволит глубже усвоить предмет изучения и более критично оценивать изучаемые вопросы. Знакомясь с особыми позициями авторов, нужно определять их схожие суждения, аргументы, выводы, а затем сравнивать их между собой и применять из них ту, которая более убедительна.

Следующим этапом работы с литературными источниками является создание конспектов, фиксирующих основные тезисы и аргументы. Можно делать записи на отдельных листах, которые потом легко систематизировать по отдельным темам изучаемого курса. Другой способ – это ведение тематических тетрадей-конспектов по одной какой-либо теме. Большие специальные работы монографического характера целесообразно конспектировать в отдельных тетрадях. Здесь важно вспомнить, что конспекты пишутся на одной стороне листа, с полями и достаточным для исправления и ремарок межстрочным расстоянием (эти правила соблюдаются для удобства редактирования). Если в конспектах приводятся цитаты, то непременно должно быть дано указание на источник (автор, название, выходные данные, № страницы). Впоследствии эта информация может быть использована при написании текста реферата или другого задания.

Таким образом, при работе с источниками и литературой важно уметь:

- сопоставлять, сравнивать, классифицировать, группировать, систематизировать информацию в соответствии с определенной учебной задачей;
- обобщать полученную информацию, оценивать прослушанное и прочитанное;
- фиксировать основное содержание сообщений; формулировать, устно и письменно, основную идею сообщения; составлять план, формулировать тезисы;
- готовить и презентовать развернутые сообщения типа доклада;
- работать в разных режимах (индивидуально, в паре, в группе), взаимодействуя друг с другом;
- пользоваться реферативными и справочными материалами;
- контролировать свои действия и действия своих товарищей, объективно оценивать свои действия;
- обращаться за помощью, дополнительными разъяснениями к преподавателю, другим студентам;
- пользоваться лингвистической или контекстуальной догадкой, словарями различного характера, различного рода подсказками, опорами в тексте (ключевые слова, структура текста, предваряющая информация и др.);
- использовать при говорении и письме перифраз, синонимичные средства, слова-описания общих понятий, разъяснения, примеры, толкования, «словотворчество»
- повторять или перефразировать реплику собеседника в подтверждении понимания его высказывания или вопроса;

- обратиться за помощью к собеседнику (уточнить вопрос, переспросить и др.);
- использовать мимику, жесты (вообще и в тех случаях, когда языковых средств не хватает для выражения тех или иных коммуникативных намерений).

15.4. Подготовка к промежуточной аттестации

При подготовке к промежуточной аттестации целесообразно:

- внимательно изучить перечень вопросов и определить, в каких источниках находятся сведения, необходимые для ответа на них;
- внимательно прочитать рекомендованную литературу;
- составить краткие конспекты ответов (планы ответов).

17. Материально-техническое обеспечение дисциплины

Таблица 18

№ п/п	Наименование специализированных аудиторий и лабораторий	Наименование оборудования
1	Лекционная аудитория	Аудио-видео комплекс
2	Аудитории для проведения групповых и практических занятий	Аудио-видео комплекс
3	Компьютерный класс	Персональные компьютеры
4	Аудитория для курсового и дипломного проектирования	Персональные компьютеры
5	Аудитория для самостоятельной работы	Компьютерная техника
6	Читальный зал	Персональные компьютеры
7	Лаборатория программной инженерии и технологий программирования	Лабораторные стенды (установки) Контрольно-измерительные приборы