

Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Санкт-Петербургский государственный университет промышленных технологий и дизайна»
(СПбГУПТД)

УТВЕРЖДАЮ
Директор ВШТЭ



Рабочая программа дисциплины

Б1.О.12 Технологии и методы программирования

Учебный план: ФГОС3++b090302Г-12_26-14.plx

Кафедра: 32 Автоматизации технологических процессов и производств

Направление подготовки:
(специальность) 09.03.02 Информационные системы и технологии

Профиль подготовки:
(специализация) Геоинформационные системы и технологии

Уровень образования: бакалавриат

Форма обучения: очная

План учебного процесса

Семестр (курс для ЗАО)		Контактная работа обучающихся		Сам. работа	Контроль, час.	Трудоё мкость, ЗЕТ	Форма промежуточной аттестации
		Лекции	Практ. занятия				
3	УП	34	51	59	36	5	Экзамен, Курсовая работа
	РПД	34	51	59	36	5	
4	УП	17	34	57	36	4	Экзамен
	РПД	17	34	57	36	4	
Итого	УП	51	85	116	72	9	
	РПД	51	85	116	72	9	

Рабочая программа дисциплины составлена в соответствии с федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.03.02 Информационные системы и технологии, утверждённым приказом Минобрнауки России от 19.09.2017 г. № 926

Составитель (и):

Препод.

Маслобоев А.Н.

От кафедры составителя:

Заведующий кафедрой автоматизации
технологических процессов и производств

Ковалев Д.А.

От выпускающей кафедры:

Заведующий кафедрой

Ковалев Д.А.

Методический отдел:

Фахретдинова А. Р.

1 ВВЕДЕНИЕ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

1.1 Цель дисциплины: Сформировать компетенции обучающегося в области использования компьютера как средства управления информацией; изучить методы программирования для овладения знаниями в области технологии программирования; подготовить обучающихся к осознанному применению, как языков программирования, так и технологий программирования.

1.2 Задачи дисциплины:

- освоение основных технологий разработки программного обеспечения;
- знакомство с методами структурного и объектно-ориентированного программирования как наиболее распространенными и эффективными технологиями разработки программных продуктов;
- изучение технологических и программных решений, используемых при разработке программного обеспечения.

1.3 Требования к предварительной подготовке обучающегося:

Предварительная подготовка предполагает создание основы для формирования компетенций, указанных в п. 2, при изучении дисциплин:

2 КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

ОПК-1: Способен применять естественнонаучные и общинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности
Знать: основные линейные и нелинейные структуры данных; наиболее важные для проектной практики алгоритмы: сортировку, поиск, обработку древовидных структур; определения структурных и числовых характеристик объектов из теории графов; алгоритмы поиска в тексте, файловые структуры.
Уметь: выполнять: анализ задачи, выбор структуры данных, разработку алгоритмов решения задачи, программную реализацию выбранных алгоритмов решения, тестирование программы, исследование и анализ алгоритмов, составление документации.
Владеть: навыками выбора оптимальных структур данных, эффективных алгоритмов обработки информации и языковых конструкций, обеспечивающих реализации типовых алгоритмов и структур данных, используемых при проектировании программ различного назначения.
ОПК-3: Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности
Знать: базовые понятия, методы и приемы программирования на языке программирования в парадигме процедурного программирования; основные базовые понятия, методы и приемы объектно-ориентированного программирования.
Уметь: писать программы на процедурном языке программирования.
Владеть: навыками программирования на процедурном языке.
ОПК-6: Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий
Знать: методы алгоритмизации и языки программирования, пригодные для практического применения в области информационных систем и технологий.
Уметь: применять методы и средства анализа функциональных требований к программному обеспечению.
Владеть: навыками анализа функциональных требований к программному обеспечению.
ПК-2: Способен разрабатывать прототипы ИС в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС
Знать: Знать - базовые понятия, методы и приемы программирования на языке программирования в парадигме процедурного программирования; основные базовые понятия, методы и приемы объектно-ориентированного программирования; методы и средства сборки модулей в программный проект в средах разработки компьютерного программного обеспечения; методы и средства проверки работоспособности программных проектов
Уметь: проводить объектную декомпозицию предметной области; проводить структурную декомпозицию программы; писать программы на процедурных и объектно-ориентированных языках программирования
Владеть: навыками анализа функциональных требований к программному обеспечению; демонстрировать способность и готовность выбора технологии и инструментальных средств, на их основе разработки, составления, отладки, тестирования и документирования программы на языках высокого уровня для различных задач

3 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Наименование и содержание разделов, тем и учебных занятий	Семестр (курс для ЗАО)	Контактная работа		СР (часы)	Форма текущего контроля
		Лек. (часы)	Пр. (часы)		
Раздел 1. Основные понятия технологии программирования, особенности программного проекта	3				
Тема 1. Основные понятия технологии программирования, особенности программного проекта. Понятие технологии программирования. Особенности промышленного программирования и "программирование на заказ". Языки программирования как средство выражения алгоритмов и как средство получения выполняемого кода. Распределение памяти в языке Си. Жизненный цикл программного обеспечения (ПО). Общая организация проекта. Модели разработки ПО. Основные технологические подходы: каскадный, каркасный, сборочный, адаптивный (экстремальное программирование). Трудности проектирование, их разновидности и преодоление. Понятие АТД. АТД, ориентированные на предметную область .		6	8	10	
Тема 2. Различные модели процесса разработки ПО. оценка осуществимости проекта, графики выполнения. Каскадная и спиралевидная модели процесса разработки ПО. Возникновение и исследование идеи. Оценка осуществимости и сложности задачи. Реальность решения задачи в заданные сроки при заданных рамочных ограничениях. Планирование: сетевой и ленточный графики – сроки, работы, ресурсы. Анализ требований и выработка спецификаций ПО. Проектирование архитектуры продукта. Выбор средств реализации. Инструментарий разработчика. Технологии RAD и готовые компоненты. Библиотеки алгоритмов и подпрограмм.		4	8	10	
Раздел 2. Основы практического программирования					

Тема 3. Базовые средства языка С Алфавит языка С. Ключевые слова языка С. Имена переменных. Типы и размеры данных. Константы. Арифметические операции. Операции сравнения и логические операции. Преобразования типов. Операции увеличения и уменьшения. Побитовые логические операции. Операции и выражения присваивания. Приоритеты в выражениях и порядок их вычисления. Операторы ввода и вывода в языке С. Директивы препроцессора.		6	9	11	
Тема 4. Основные операторы языка С Операторы ветвления. Оператор if. Полная и сокращенная формы записи оператора. Переключатель switch. Оператор break. Операторы циклов. Цикл с заранее заданным количеством повторений for. Цикл с предусловием while. Цикл с постусловием do-while. Массивы данных. Строки. Ввод строк. Вывод строк. Ввод и вывод символов. Описание функции. Вызов функции. Функция main(). Локальные и глобальные переменные. Передача переменных в функцию.		6	9	8	
Раздел 3. Структурное программирование					
Тема 5. Указатели в языке С++. Функции в языке С++ Указатель на функцию. Указатель на объект. Указатель на void. Динамическая память и динамические переменные. Инициализация указателей. Операции с указателями (операция разыменования, арифметические операции). Ссылки. Параметры функции. Рекурсивные функции. Перегрузка функций. Шаблоны функций. Функции стандартной библиотеки.		6	9	10	
Тема 6. Типы данных, определяемые пользователем в С++. Динамические структуры данных в С++. Переименование типов (typedef). Перечисления (enum). Структуры (struct). Поля структуры. Описание и инициализация структур. Доступ к полям структуры. Битовые поля. Объединения (union). Ограничения, накладываемые на объединения. Линейные списки. Однонаправленные, двунаправленные и кольцевые списки. Основные операции со списками. Стеки. Очереди. Бинарные деревья. Реализация динамических структур с помощью массивов.		6	8	10	
Итого в семестре (на курсе для ЗАО)		34	51	59	
Консультации и промежуточная аттестация (Экзамен, Курсовая работа)		2,5		33,5	

Раздел 4. Объектно-ориентированное программирование				
Тема 7. Классы в языке C++. Описание классов. Описание объекта. Указатель this. Конструкторы. Основные свойства конструкторов. Конструктор копирования. Статические элементы класса (статические поля и статические методы). Дружественные функции. Правила описания и особенности дружественных функций. Дружественные классы. Деструкторы. Перегрузка операций. Указатели на элементы классов. Рекомендации по составу классов.	4	8	12	
Тема 8. Наследование в языке C++ Базовые классы. Ключи доступа. Простое наследование. Порядок вызова конструкторов при наследовании. Правила наследования деструкторов. Виртуальные методы, правила их описания и использования. Чисто виртуальные методы. Механизм позднего связывания. Абстрактные классы. Множественное наследование. Отличия структур и объединений от классов.	4	8	12	
Тема 9. Шаблоны классов Создание шаблонов классов. Синтаксис описания шаблона. Методы шаблона класса. Правила описания шаблонов. Использование шаблонов классов. Специализация шаблонов классов. Специализация метода. Специализация класса. Достоинства и недостатки шаблонов классов.	3	6	11	
Раздел 5. Исключительные ситуации. Стандартные библиотеки				
Тема 10. Обработка исключительных ситуаций Общий механизм обработки исключений. Стек вызовов. Раскручивание стека. Синтаксис исключений. Контролируемый блок. Генерация исключения. Обработчики исключений. Перехват исключений. Список исключений функции. Исключения в конструкторах и деструкторах. Иерархии исключений.	3	6	11	

Тема 11. Работа со стандартной библиотекой					
Стандартные потоки. Форматирование данных. Флаги и форматирующие методы. Манипуляторы. Методы обмена с потоками. Ошибки потоков. Файловые потоки. Строковые потоки. Потоки и типы, определенные пользователем. Последовательные контейнеры. Векторы. Очереди. Двусторонние очереди. Списки. Стеки. Очереди с приоритетами. Ассоциативные контейнеры. Словари. Словари с дубликатами. Множества. Множества с дубликатами. Битовые множества.		3	6	11	
Итого в семестре (на курсе для ЗАО)		17	34	57	
Консультации и промежуточная аттестация (Экзамен)		2,5		33,5	
Всего контактная работа и СР по дисциплине		141		183	

4 КУРСОВОЕ ПРОЕКТИРОВАНИЕ

4.1 Цели и задачи курсовой работы (проекта): Целью курсовой работы является закрепление знаний, полученных в процессе изучения дисциплины, и развитие практических навыков в процессе разработке полноценного программного продукта.

Задачей курсовой работы является создание современного работоспособного приложения с использованием современных технологий структурного и объектно-ориентированного программирования.

4.2 Тематика курсовой работы (проекта): Список тем для курсовой работы:

1. Разработка простого текстового редактора. В нем должны быть предусмотрены возможности для создания и сохранения текстовых файлов, вырезания, копирования, вставки и удаления фрагментов текста, поиска и замены текста, вставки в документ даты и времени.

дублирования тем курсовой работы в рамках одной учебной группы.

Список тем для курсовой работы:

1. Разработка простого текстового редактора. В нем должны быть предусмотрены возможности для создания и сохранения текстовых файлов, вырезания, копирования, вставки и удаления фрагментов текста, поиска и замены текста, вставки в документ даты и времени.

2. Разработка приложения для автоматического рерайтинга текста. Под рерайтингом понимается обработка исходных текстовых материалов для дальнейшего использования. За основу берется уже написанный текст, который переписывается своими словами с сохранением смысла.

3. Разработка инженерного калькулятора. Данный калькулятор, помимо выполнения основных арифметических операций, должен обеспечивать вычисление степени числа, десятичного и натурального логарифма, экспоненты и факториала числа.

4. Разработка калькулятора для программиста. Данный калькулятор должен выполнять перевод чисел в двоичную, восьмеричную, десятичную и шестнадцатеричную системы счисления, а также выполнять основные арифметические операции в этих системах.

5. Разработка финансового калькулятора. Данный калькулятор должен обеспечивать перевод денежных сумм из рублей в доллары, евро и юани, и обратный перевод сумм из этих валют в рубли. Также в калькуляторе должна быть предусмотрена возможность вычисления текущей суммы банковского вклада, исходя из начальной величины вклада, процентов по вкладу и срока действия договора с банком.

6. Разработка логического калькулятора. Калькулятор должен выполнять логические операции «NOT», «AND», «OR», «XOR», «NOR», «NAND» а также операции сдвига битов.

7. Разработка статистического калькулятора. Данный калькулятор для заданного ряда чисел должен находить их среднее арифметическое, среднее геометрическое, моду, медиану, среднеквадратичное отклонение, дисперсию, сумму элементов ряда, наибольший и наименьший элемент.

8. Разработка калькулятора «Площадь геометрических фигур». Данный калькулятор должен:

а) находить площадь треугольника по 3 длинам сторон, по 2 сторонам и углу между ними; по полупериметру и радиусу вписанной окружности;

б) площадь квадрата по стороне и диагонали квадрата;

в) площадь прямоугольника;

г) площадь параллелограмма по сторонам и углу между ними, а также по стороне и высоте;

д) площадь ромба;

е) площадь трапеции;

ж) площадь четырехугольника;

з) площадь круга;

и) площадь эллипса.

9. Разработка калькулятора «Аналитическая геометрия». Данный калькулятор точек должен находить:

а) длину отрезка по заданным координатам;

- б) середину отрезка по заданным координатам;
- в) уравнение прямой, проходящей через две точки по заданным координатам;
- г) точку пересечения двух прямых по заданным уравнениям;
- д) угол между прямыми по заданным уравнениям;
- е) расстояние от точки до прямой на плоскости по координатам точки и уравнению прямой;
- ж) уравнение плоскости по координатам трех точек.

10. Разработка калькулятора «Комплексные числа». Данный калькулятор должен

- а) выполнять операции сложения, вычитания, умножения и деления комплексных чисел;
- б) вычислять модуль комплексного числа.

11. Разработка калькулятора «Теория чисел». Данный калькулятор должен:

- а) находить наибольший общий делитель для двух введенных чисел;
- б) наименьшее общее кратное для двух введенных чисел;
- в) проверять, является ли введенное число простым или нет;
- г) выполнять разложение введенного числа на простые множители;
- д) проверять, являются ли два введенных числа дружественными.

12. Разработка калькулятора «Вычисления с дробями». Данный калькулятор должен:

- а) складывать, вычитать, умножать и делить дроби;
- б) вычисления с обыкновенной и десятичными дробями;
- в) преобразование десятичной дроби в обыкновенную дробь;
- г) преобразование неправильных дробей в смешанные числа;
- д) сокращение дробей;
- е) приведение дробей к общему знаменателю.

13. Разработка клавиатурного тренажера. В тренажере должна быть предусмотрена возможность ввода текста для тренировки пользователя из файла. Оценка работы пользователя должна производиться по 3 параметрам:

- а) количество ошибок;
- б) скорость набора;
- в) ритмичность набора.

14. Разработка приложения «Записная книжка». В приложении должна быть предусмотрена возможность работы с произвольным числом записей, поиск записи по определенному признаку (по фамилии, дате рождения или номеру телефона), добавления и удаления записей, сортировки по перечисленным выше признакам.

15. Разработка приложения «Студенческая группа». В приложении должна быть предусмотрена возможность работы с переменным числом студентов, поиска студента по определенному признаку (по фамилии,

дате рождения или номеру телефона), добавления и удаления записей, сортировки по перечисленным выше признакам.

16. Разработка приложения «Библиотека». В приложении должна быть предусмотрена возможность работы с произвольным числом книг, поиска книги по определенному признаку (по фамилии автора или году издания), добавления и удаления книг в библиотеку, сортировки книг по разным полям.

17. Разработка приложения «Геометрические фигуры». В программе должна быть предусмотрена опция выбора фигуры: круг, квадрат или прямоугольник. Для каждого из перечисленных объектов должна быть задана возможность его создания по заданным координатам, перемещения на плоскости, изменения размеров и поворота на заданный угол.

18. Разработка приложения «Прямоугольник». Данное приложение должно работать с прямоугольниками, со сторонами, параллельными осям координат. В приложении должна быть предусмотрена возможность перемещения прямоугольников на плоскости, изменение размеров, построение наименьшего прямоугольника, содержащего два заданных прямоугольника, и прямоугольника, являющегося общей частью (пересечением) двух прямоугольников.

19. Разработка приложения «Тригонометрические функции». Приложение должно обеспечить возможность выбора одной из тригонометрических функций (синус, косинус, тангенс, котангенс, секанс, косеканс) и для выбранной функции построить ее график.

20. Разработка приложения «Вектор». Данное приложение должно работать с объектами-векторами, задаваемыми координатами концов в трехмерном пространстве. Приложение должно выполнять операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами.

21. Разработка приложения «Массив». Данное приложение должно работать с одномерными массивами целых чисел. В приложении должны быть предусмотрены возможности обращения к отдельному элементу массива с контролем выхода за пределы массива, задания произвольных границ индексов при создании объекта, выполнения операций поэлементного сложения и вычитания массивов с одинаковыми границами индексов, умножения и деления всех элементов массива на скаляр, вывода на экран элемента массива по заданному индексу, вывода на экран всего массива.

22. Разработка приложения «Матрица». Приложение должно реализовывать следующие операции над матрицами:

- а) сложение, вычитание, умножение, деление (умножение и деление как на другую матрицу, так и на число);
- б) операции сравнения на равенство или неравенство;
- в) операции вычисления обратной и транспонированной матрицы, операцию возведения в степень;
- г) вычисление детерминанта матрицы;
- д) проверка матрицы (является ли она квадратной, диагональной, нулевой, симметрической).

23. Разработка приложения «Полином». Приложение должно работать с многочленами от одной переменной, которые задаются степенью многочлена и массивом коэффициентов. В программе должна быть предусмотрена возможность вычисления значения многочлена для заданного аргумента, операций сложения, вычитания и умножения многочленов с получением нового объекта-многочлена, вывода на экран описания многочлена.

24. Разработка приложения «Строка». Приложение должно работать с одномерными массивами строк фиксированной длины. В приложении должна быть предусмотрена возможность обращения к отдельным строкам массива по индексам, контроль выхода за пределы массива, выполнение операции поэлементного сцепления двух массивов с образованием нового массива, слияния двух массивов с исключением повторяющихся элементов, вывод на экран элемента массива по заданному индексу и всего массива.

4.3 Требования к выполнению и представлению результатов курсовой работы (проекта):

Курсовая работа должна быть выполнена обучающимся в течение семестра обучения.

Курсовая работа должна быть представлена в виде программной документации и созданного обучающимся программного продукта в виде исполняемого файла. Программная документация должна быть оформлена в соответствии с действующими стандартами и состоять из 3-х разделов

а) задание на курсовую работу (выдается преподавателем)

б) описание программного продукта

в) исходные коды модулей программного продукта.

При оценивании уровня курсовой работы основными критериями являются:

а) достижение цели работы

б) уровень изложения и интерпретации результатов работы

в) правильность и полнота ответов на вопросы преподавателя

5 БАЛЛЬНО-РЕЙТИНГОВАЯ СИСТЕМА ОЦЕНИВАНИЯ

Использование балльно-рейтинговой системы не предусмотрено.

6. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ, КРИТЕРИЕВ, СИСТЕМЫ И СРЕДСТВ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ

6.1 Показатели оценивания

Код компетенции	Показатели оценивания результатов обучения
ОПК-1	Обладает знаниями по основным структурам данных языков С и С++, наиболее эффективным методам поиска и сортировки данных. Способен самостоятельно выбрать оптимальный алгоритм для решения поставленной задачи. Готов к использованию изученных алгоритмов и структур данных для практического использования в своей профессиональной деятельности.
ОПК-3	Обладает знаниями об основных принципах структурного и объектно-ориентированного программирования, используемых в языках С и С++. Способен к самостоятельному созданию работоспособных приложений на языках С и С++. Готов к использованию полученных навыков в области технологий и методов программирования в своей профессиональной деятельности.
ОПК-6	Обладает необходимыми знаниями языков программирования С и С++ для их использования применительно к современным информационным технологиям. Способен самостоятельно оценить возможности программного обеспечения для решения практических задач. Готов к проведению квалифицированного анализа программного обеспечения и выполнения, в случае необходимости, его оптимизации.
ПК-2	Обладает необходимыми знаниями об основных парадигмах программирования, включая процедурное, объектно-ориентированное и функциональное программирование. Способен к самостоятельному решению задач по структурной декомпозиции программы и объектно декомпозиции предметной области. Готов к выполнению всех видов деятельности, необходимых для обеспечения жизненного цикла программного продукта.

6.2 Система и критерии оценивания

Шкала оценивания	Критерии оценивания сформированности компетенций	
	Устное собеседование	Письменная работа
5 (отлично)	Обучающийся показывает всестороннее и глубокое знание теоретических основ дисциплины,	Полное и всесторонне рассмотрение вопросов, свидетельствующее о значительной самостоятельной работе с

	свободно ориентируется в основных понятиях, терминах и определениях при ответе; знаком с дополнительной литературой; способен проработать научно-исследовательскую литературу по темам дисциплины и грамотно изложить материал. Качество исполнения всех элементов практического задания полностью соответствует предъявляемым требованиям.	источниками информации. Качество исполнения всех элементов работы соответствует требованиям, содержание полностью соответствует заданию. Полученные результаты представлены на основании изучения и анализа исследуемой темы. Даны полные выводы и ответы на поставленные вопросы. Работа представлена к защите в срок.
4 (хорошо)	Обучающийся показывает знание теоретических основ дисциплины, свободно ориентируется в основных понятиях, терминах и определениях при ответе; знаком с дополнительной литературой; способен проработать научно-исследовательскую литературу по темам дисциплины и грамотно изложить материал, но допускает ошибки при ответах на дополнительные вопросы преподавателя. Практическое задание выполнено в соответствии с поставленной задачей. Имеются отдельные несущественные ошибки или отступления от правил оформления.	Работа выполнена в соответствии с заданием. Имеются отдельные несущественные ошибки в работе или ответах на поставленные при защите вопросы, могут иметь место отступления от правил оформления работы или нарушены сроки сдачи работы.
3 (удовлетворительно)	Обучающийся показывает неполное знание теоретических основ дисциплины, ориентируется в основных понятиях, терминах и определениях при ответе, но не знаком с дополнительной литературой; может проработать научно-исследовательскую литературу по темам дисциплины, но не может грамотно и четко изложить материал, допускает ошибки при ответах на дополнительные вопросы преподавателя. Практическое задание выполнено полностью, но с существенными ошибками. При этом нарушены правила оформления.	Задание выполнено полностью, но в работе есть отдельные существенные ошибки, присутствуют неточности в ответах. Качество работы низкое, либо работа представлена с опозданием.
2 (неудовлетворительно)	Обучающийся не знает теоретических основ дисциплины, не способен проработать научно-исследовательскую литературу по темам дисциплины, не может грамотно и четко изложить материал, допускает ошибки при ответах на дополнительные вопросы преподавателя.	Содержание работы полностью не соответствует заданию. Отсутствуют один или несколько обязательных элементов задания. Допущены многочисленные грубые ошибки при выполнении. Нарушение правил оформления, неспособность ответить на дополнительные вопросы. Нарушение сроков сдачи работы.

6.3 Типовые контрольные задания и/или иные материалы, используемые для оценки результатов обучения

6.3.1 Типовые контрольные вопросы

№ п/п	Формулировки вопросов
Семестр 3	
1	Понятие технологии программирования
2	Основные требования к программному обеспечению
3	Понятие о жизненном цикле программного обеспечения
4	Каскадная модель жизненного цикла программного обеспечения
5	Инкрементная модель жизненного цикла программного обеспечения
6	Спиральная модель жизненного цикла программного обеспечения
7	Эволюционная модель жизненного цикла программного обеспечения

8	Стандартизация модели жизненного цикла программного обеспечения
9	Понятие зрелости процессов создания программного обеспечения. Модель оценки зрелости CMM
10	Метод функционального моделирования SADT
11	Метод моделирования процессов IDEF3
12	Основные принципы построения объектной модели
13	Алфавит и ключевые слова языка C
14	Переменные в языке C. Базовые типы переменных
15	Структура программы на языке C. Препроцессор языка C.
16	Операторы ввода и вывода данных в языке C
17	Арифметические и логические операции в языке C. Приоритеты при их выполнении
18	Условный оператор в языке C.
19	Циклы с предусловием и постусловием в языке C
20	Цикл с заранее заданным количеством повторений в языке C
21	Указатели в языке C++.
22	Ссылки в языке C++
23	Списки в языке C++
24	Функции в языке C++
25	Бинарные деревья в языке C++
Семестр 4	
26	Ввод и вывод данных в языке C++
27	Описание класса в языке C++
28	Объекты в языке C++
29	Конструкторы в языке C++
30	Статические элементы класса в языке C++
31	Дружественные функции в языке C++.
32	Дружественные классы в языке C++.
33	Деструкторы в языке C++
34	Перегрузка унарных операций в языке C++
35	Перегрузка бинарных операций в языке C++.
36	Перегрузка операций присваивания в языке C++.
37	Перегрузка операции приведения типа в языке C++.
38	Перегрузка операции вызова функции в языке C++.
39	Перегрузка операции индексирования в языке C++.
40	Указатели на элементы классов в языке C++
41	Ключи доступа в языке C++.
42	Простое наследование в языке C++.
43	Множественное наследование в языке C++.
44	Абстрактные классы в языке C++.
45	Создание и использование шаблонов классов в языке C++
46	Специализация шаблонов классов в языке C++
47	Иерархия исключений в языке C++.
48	Динамическое определение типа в языке C++
49	Файловые потоки в языке C++
50	Строковые потоки в языке C++.

6.3.2 Типовые тестовые задания

не предусмотрено

6.3.3 Типовые практико-ориентированные задания (задачи, кейсы)

1. Разработать на языке С приложение, которое по введенным длинам сторон треугольника определяет, может ли существовать такой треугольник. В случае положительного ответа вычислить площадь треугольника по формуле Герона.
2. На языке С++ разработать приложение, которое находит все совершенные числа в диапазоне от 1 до n, где n - натуральное число, введенное пользователем.
3. На языке С++ разработать приложение, которое определяет, являются ли два введенных натуральных числа дружественными.

6.4 Методические материалы, определяющие процедуры оценивания результатов обучения

6.4.1 Условия допуска обучающегося к промежуточной аттестации и порядок ликвидации академической задолженности

Проведение промежуточной аттестации регламентировано локальным нормативным актом СПбГУПТД «Положение о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся»

6.4.2 Форма проведения промежуточной аттестации по дисциплине

Устная ☐ + Письменная ☐ + Компьютерное тестирование ☐ Иная ☐

6.4.3 Особенности проведения промежуточной аттестации по дисциплине

- Время на подготовку ответа на экзамене - 30 минут;
- Время, отводимое на защиту курсовой работы, не должно превышать 15 мин, включая краткий доклад по результатам курсовой работы и ответы на вопросы
- Возможность пользоваться справочными материалами, калькулятором.

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

7.1 Учебная литература

Автор	Заглавие	Издательство	Год издания	Ссылка
7.1.1 Основная учебная литература				
Немцова Т.И., Голова С.Ю., Терентьев А.И. и др.	Программирование на языке высокого уровня. Программирование на языке С++	Москва: Форум	2021	https://ibooks.ru/reading.php?short=1&productid=361544
Терехов, А. Н.	Технология программирования	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2020	http://www.iprbookshop.ru/97587.html
Борисенко, В. В.	Основы программирования	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2020	http://www.iprbookshop.ru/97568.html
7.1.2 Дополнительная учебная литература				
Александров, Э. Э., Афонин, В. В.	Программирование на языке С в Microsoft Visual Studio 2010	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2021	http://www.iprbookshop.ru/102050.html
Зыков, С. В.	Введение в теорию программирования. Объектно-ориентированный подход	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2021	http://www.iprbookshop.ru/102007.html

Баженова, И. Ю., Сухомлин, В. А.	Введение программирование	в	Москва: Интернет- Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2020	http://www.iprbookshop.ru/97539.html
-------------------------------------	------------------------------	---	--	------	---

7.2 Перечень профессиональных баз данных и информационно-справочных систем

Электронно-библиотечная система IPRbooks [Электронный ресурс]. URL: <http://www.iprbookshop.ru/>
 Электронная библиотека ВШТЭ СПб ГУПТД [Электронный ресурс]. URL: <http://nizrp.narod.ru>
 Электронно-библиотечная система «Айбукс» [Электронный ресурс]. URL: <https://www.ibooks.ru/>
 Информационная система «Единое окно доступа к образовательным ресурсам. [Электронный ресурс].

7.3 Перечень лицензионного и свободно распространяемого программного обеспечения

Microsoft: Office Standard 2016 Russian OLP NL AcademicEdition
 Microsoft: WIN HOME 10 Russian OLPNL AcademicEdition Legalization GetGenuine

7.4 Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Аудитория	Оснащение
Лекционная аудитория	Мультимедийное оборудование, специализированная мебель, доска
Компьютерный класс	Мультимедийное оборудование, компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду