

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 31.08.2019
Уникальный программный ключ:
8db180d1a3f02ac9e60521a517741755e1861d0

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский политехнический университет»

УТВЕРЖДАЮ



Декан факультета
информационных технологий
Филиппович А.Ю.
“01” сентября 2019 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Теория вычислительных процессов и языков программирования»

Направление подготовки
09.03.01 Информатика и вычислительная техника

Образовательная программа (профиль подготовки)
« Программное обеспечение информационных систем»

Квалификация (степень) выпускника
бакалавр

Форма обучения
заочная

Год приема - 2019

Москва 2019 г.

Рабочая программа дисциплины составлена в соответствии с федеральным образовательным стандартом высшего образования – магистратуры по направлению подготовки 09.03.01 Информатика и вычислительная техника

Рабочая программа утверждена на заседании кафедры Прикладная информатика "28" августа 2019 г (Протокол №1)

Заведующий кафедрой «Прикладная информатика»:

_____ / С. В. Суворов /

Согласовано:

Руководитель образовательной программы:

_____ / С. В. Суворов /

Разработчик

Доцент кафедры "Прикладная информатика"

/В. Г. Евтихов/

1. Цели освоения дисциплины

Основные цели дисциплины «Теория вычислительных процессов и языков программирования»: приобретение обучающимися фундаментальных знаний в области теории вычислительных процессов и языков программирования; формирование у обучающихся навыков практического применения приобретенных знаний.

Основные задачи дисциплины: ознакомление обучающихся с основными результатами математической теории формальных языков, автоматов и вычислений; ознакомление обучающихся с примерами применения теории формальных языков и автоматов для построения компиляторов и реализации языков программирования; освоение обучающимися элементов теории сложности для осуществления оценки принципиальных возможностей разрабатываемого программного обеспечения; освоение обучающимися основных методов разработки программного обеспечения для проверки систем, характеризующихся конечным числом различных состояний.

2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «Теория вычислительных процессов и языков программирования» относится к числу профессиональных учебных дисциплин части формируемой участниками образовательных отношений базового цикла основной образовательной программы бакалавриата.

Для изучения данной дисциплины необходимо предварительное изучение обучающимися логически и содержательно-методически взаимосвязанных с ней дисциплин: «Программирование», «Математический анализ», «Алгебра», «Операционные системы», «Вычислительная математика», «Структуры и алгоритмы обработки данных».

Знания, навыки и умения, приобретенные в результате прохождения курса, также могут быть востребованы студентами при подготовке выпускной квалификационной работы бакалавра.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

В процессе изучения дисциплины студент осваивает компетенцию ПК-1, характеризующую следующими основными компонентами. В результате освоения дисциплины обучающимися должны быть достигнуты следующие результаты обучения (как этап формирования соответствующей компетенции):

Код компетенции	В результате освоения образовательной программы обучающийся должен обладать	Перечень планируемых результатов обучения по дисциплине
ПК-1	способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина»	знать: - методы и основные этапы трансляции; - основные положения теории вычислительных процессов и структур, их применение при реализации языков программирования и создании прикладных информационных систем; - принципы и способы технической реализации моделей процессов и структур; - теоретические основы синтаксического анализа, перевода, компиляции и машин Тьюринга. уметь: - определять различными способами формальные языки, корректно и свободно преобразовывать представления языка, применять теорию автоматов для

		<p>решения практических задач; - строить конечные автоматы по заданным грамматикам, минимизировать автоматы, устранять левую рекурсию, приводить грамматику к нормальной форме Хомского, применять полученные знания для решения конкретных прикладных задач; - разрабатывать компоненты программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования.</p> <p>владеть: - навыками использования методик теории формальных языков, автоматов и вычислений при решении практических задач; - основными навыками вычислительной теории и построения компиляторов; - навыками осваивать методики использования программных средств для решения практических задач.</p>
--	--	---

4. Структура и содержание дисциплины

Дисциплина «Теория вычислительных процессов и языков программирования» для студентов заочной формы обучения преподается в 7 семестре 4 курса.

Общая трудоемкость дисциплины составляет 5 зачетных единиц, 180 часов.

Аудиторные занятия по заочной форме обучения составляют 35 часов.

Из них: 8 часов лекций, 27 часов лабораторных работ.

Самостоятельная работа студента составляет 145 часов.

Форма контроля на пятом курсе в 9 семестре: зачет и экзамен.

4.1. Содержание разделов дисциплины

Дисциплина «Теория вычислительных процессов и языков программирования» включает следующие разделы:

Тема 1. Теория схем программ.

Программы и схемы программ. Стандартные схемы программ. Базис класса стандартных схем программ. Графовая форма стандартной схемы. Линейная форма стандартной схемы. Интерпретация стандартных схем программ. Свойства и виды стандартных схем программ. Эквивалентность, тотальность, пустота, свобода. Свободные интерпретации. Согласованные свободные интерпретации. Логико-термальная. Моделирование стандартных схем программ. Одноленточный автомат. Многоленточный автомат. Двухголовочный автомат. Двоичный двухголовочный автомат. Моделирование двоичного автомата стандартной схемой.

Тема 2. Семантическая теория программ.

Формализация семантики. Операционная семантика. Денотационная семантика. Аксиоматическая семантика. Верификация программ. Методы доказательства правильности. Доказательство частичной корректности программы методом индуктивных утверждений. Методика верификации программы. Анализ завершения программы. Анализ незавершения программы. Автоматизация верификации программ.

Тема 3. Асинхронные процессы.

Формальное определение асинхронного процесса. Подклассы асинхронных процессов. Структурирование асинхронного процесса. Интерпретация асинхронного процесса. Асинхронный процесс как метамодель. Диаграммы переходов. Сигнальные графы.

Тема 4. Взаимодействие процессов.

Классификация вычислительных процессов. Классические задачи взаимодействия асинхронных процессов. Задача взаимного исключения. Задача «производитель-потребитель».

Задача «читатели-писатели». Задача «обедающие философы». Проблема тупиков. Средства синхронизации взаимодействующих вычислительных процессов. Блокировка памяти. Операции типа «проверка и установка». Семафорные механизмы. Мониторы Хоара. Почтовые ящики. Конвейеры и очереди сообщений.

Тема 5. Модели вычислительных процессов.

Вычислительные схемы. Модель пространства состояний системы. Модель Холта. Описание модели повторно используемых ресурсов. Обнаружение тупика посредством редукции графа повторно используемых ресурсов. Методы обнаружения тупика по наличию замкнутой цепочки запросов.

Тема 6. Сети Петри.

Описание модели. Формальное определение сети Петри. Правила функционирования сетей Петри. Дерево разметок сети. Свойства сетей Петри. Анализ сетей Петри. Матричный подход к анализу сетей Петри. Моделирование систем на основе сетей Петри. Моделирование последовательных процессов. Моделирование взаимоисключения параллельных процессов. Моделирование взаимодействия типа «производитель – потребитель».

4.2. Темы лабораторных работ

1. Алфавиты, языки и цепочки. Построение грамматик для различных, в том числе и контекстно-зависимых, языков;

2. Написание регулярных выражений. Переход между регулярными выражениями, детерминированными конечными автоматами, недетерминированными конечными автоматами и праволинейными грамматиками. Минимизация конечных автоматов;

3. Построение контекстно-свободных грамматик, удаление бесполезных и недостижимых символов. Устранение цепных и эпсилон-правил. Устранение левой рекурсии. Построение автоматов с магазинной памятью для КС-языков;

4. Приведение грамматик к нормальной форме Хомского. Построение машин Тьюринга. Вычислимость и невычислимость.

Удельный вес занятий, проводимых в интерактивных формах, определен главной целью образовательной программы, особенностью контингента обучающихся и содержанием дисциплины «Теория вычислительных процессов и языков программирования» и в целом по дисциплине составляет 25% аудиторных занятий. Занятия лекционного типа составляют около 23% от объема аудиторных занятий.

5. Образовательные технологии

Методика преподавания дисциплины «Теория вычислительных процессов и языков программирования» и реализация компетентностного подхода в изложении и восприятии материала предусматривает использование следующих активных и интерактивных форм проведения групповых, индивидуальных, аудиторных занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся:

- подготовка к лекциям и к выполнению лабораторных работ;
- подготовка и разработка лабораторных работ;
- подготовка к самостоятельной работе студента.

Лекционные занятия проводятся в форме изложения преподавателем учебного материала в течение занятий согласно календарно-тематическому плану и с использованием видеопроектора и локальной сети компьютерных классов ав.1201, ав.1202.

Лабораторные занятия выполняются студентами с использованием оборудования локальной сети компьютерных классов, оснащенных выходом в Интернет. Объем лекционных занятий соответствует требованиям ФГОС по направлению 09.03.01 «Информатика и вычислительная техника». Большое значение для изучения теоретического материала и приобретения практических навыков имеет самостоятельная работа, на которую отводится 160

часа. При самостоятельной работе студент получает навыки поиска, анализа и систематизации требуемой для выполнения лабораторных работ информации, изучает теоретические и практические аспекты разработки алгоритмов, моделей и программ, а также применения инструментальных средств для реализации практических задач в предметной области.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов.

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

- подготовка к выполнению лабораторных работ и защита их результатов
- проведение зачета
- проведение экзамена.

Примеры вопросов для самопроверки, вопросов к зачету и экзамену приведены в приложении.

6.1. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

6.1.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы.

В результате освоения дисциплины формируется следующая компетенция:

Код компетенции	В результате освоения образовательной программы обучающийся должен обладать
ПК-1	способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина»

В процессе освоения образовательной программы данная компетенция, в том числе ее отдельные компоненты, формируются поэтапно в ходе освоения обучающимися дисциплины, практик в соответствии с учебным планом и календарным графиком учебного процесса.

6.1.2. Описание показателей и критериев оценивания компетенций, формируемых по итогам освоения дисциплины, описание шкал оценивания

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине «Теория вычислительных процессов и языков программирования».

ПК 1 - способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина»				
Показатель	Критерии оценивания			
	2	3	4	5

<p>знать: - методы и основные этапы трансляции; - основные положения теории вычислительных процессов и структур, их применение при реализации языков программирования и создании прикладных информационных систем; - принципы и способы технической реализации моделей процессов и структур; - теоретические основы синтаксического анализа, перевода, компиляции и машин Тьюринга.</p>	<p>Обучающийся не знает основы дискретной математики и математические основы современной вычислительной техники, не имеет представлений об общих структурах и не понимает определение формального языка;</p>	<p>Обучающийся знает и понимает определения регулярного языка и грамматики, может привести примеры, знает понятие асимптотики функций, общие структуры данных и алгоритмы работы с ними, знает приложения теории формальных языков для компиляторов.</p>	<p>Обучающийся знает и понимает определения контекстно-свободного языка и грамматики, может привести примеры, знает основные определения и утверждения теории множеств и математического анализа в контексте теории языков, автоматов и вычислений, знает приложения теории автоматов для компиляторов</p>	<p>Обучающийся знает теоретические основы дискретной математики и математического анализа в контексте теории языков, автоматов и вычислений, знает математические основы современной вычислительной техники, знает и понимает определение машины Тьюринга, может формализовать понятие разрешимых и неразрешимых проблем посредством ЭВМ.</p>
<p>уметь: - определять различными способами формальные языки, корректно и свободно преобразовывать представления языка, применять теорию автоматов для решения практических задач; - строить конечные автоматы по заданным грамматикам, минимизировать автоматы, устранять левую рекурсию, приводить грамматику к нормальной форме Хомского, применять полученные знания для решения конкретных прикладных задач;</p>	<p>Обучающийся не умеет формализовать в терминологии теории языков, автоматов и вычислений условия учебных упражнений из тематической литературы, не умеет интерпретировать оценки временной и емкостной эффективности алгоритмов</p>	<p>Обучающийся умеет формализовать в терминологии теории языков, автоматов и вычислений условия учебных упражнений из тематической литературы, умеет интерпретировать оценки временной сложности алгоритмов теории формальных языков и автоматов и применять полученные знания для решения конкретных прикладных задач.</p>	<p>Обучающийся умеет обоснованно предложить направления решений учебных упражнений из тематической литературы, умеет проводить анализ эффективности алгоритмов теории формальных языков и автоматов, обосновывать выбор инструментальных средств для решения конкретных задач,.</p>	<p>Обучающийся умеет свободно решать учебные упражнения из тематической литературы, умеет проводить анализ эффективности разрабатываемых на основе теории автоматов алгоритмов, обосновывать выбор представления и обработки данных для решения поставленной задачи, применять инструментальные средства и технологии программирования для решения конкретных задач; разрабатывать модели различных классов систем.</p>

<p>- разрабатывать компоненты программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования.</p>				
<p>владеть: - навыками использования методик теории формальных языков, автоматов и вычислений при решении практических задач; - основными навыками вычислительной теории и построения компиляторов; - навыками осваивать методики использования программных средств для решения практических задач.</p>	<p>Обучающийся не владеет навыками выделения основных этапов процесса компиляции и использования программных средств для решения практических задач</p>	<p>Обучающийся владеет навыками формулировать собственные определения и утверждения при решении учебных задач, владеет методами автоматического определения синтаксиса и использования программных средств для решения практических задач</p>	<p>Обучающийся владеет навыками логически доказывать и проверять собственные определения и утверждения при решении учебных задач, владеет методами описания и реализации языков программирования и использования программных средств для решения практических задач</p>	<p>Обучающийся владеет способами нахождения и методами доказательств решений, владеет навыками работы с инструментальными средствами для создания компиляторов, навыками разработки рекурсивных функций, навыками использования программных средств для решения практических задач</p>

6.2. Шкалы оценивания результатов промежуточной аттестации и их описание.

Форма промежуточной аттестации на 5 курсе в 9 семестре: зачет

Промежуточная аттестация обучающихся в форме зачёта проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине (модулю), при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине (модулю) проводится преподавателем, ведущим занятия по дисциплине (модулю) методом экспертной оценки. Перечень тематик лабораторных работ представлен в пункте 4.2. данной рабочей программы.

Защита лабораторных работ проводится в форме зачета. При этом определяется уровень практических навыков студента и соответствие работы предъявляемым к ней требованиям. Основными критериями оценки работы являются качество ее выполнения, а также эффективность применения методов разработки математических, алгоритмических и программных разработок. Успешная защита лабораторных работ является условием допуска студента к сдаче зачета по дисциплине «Теория вычислительных процессов и языков программирования».

По результатам защиты отчета по лабораторным выставляется оценка: «зачтено», «не зачтено». Студенты, работа которых не была зачтена, не допускаются до итогового зачета по дисциплине «Теория вычислительных процессов и языков программирования».

По итогам промежуточной аттестации по дисциплине (модулю) выставляется оценка «зачтено» или «не зачтено».

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой по дисциплине «Теория вычислительных процессов и языков программирования» – прошли промежуточный контроль, выполнили лабораторные работы

Шкала оценивания	Описание
Зачтено	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Не зачтено	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

Форма промежуточной аттестации на 5 курсе в 9 семестре: экзамен

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине, при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине проводится преподавателем, ведущим занятия по дисциплине методом экспертной оценки. По итогам промежуточной аттестации по дисциплине выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой по дисциплине «Теория вычислительных процессов и структур» и получившие зачет.

Шкала оценивания	Описание
Отлично	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Хорошо	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент в основном демонстрирует соответствие знаний, умений, навыков приведенным в

	таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены некоторые ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Удовлетворительно	Студент демонстрирует удовлетворительное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются умеренные ошибки, проявляется неполное наличие знаний, умений, навыков по ряду показателей, студент испытывает затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
Неудовлетворительно	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

Фонды оценочных средств представлены в приложении 1 к рабочей программе.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)

Направление подготовки: 09.03.01 Информатика и вычислительная техника
ОП (профиль): «Программное обеспечение средств вычислительной техники и
автоматизированных систем»
Форма обучения: заочная
Вид профессиональной деятельности: проектно-конструкторская деятельность

Кафедра: Прикладная информатика

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ДИСЦИПЛИНЕ**

Теория вычислительных процессов и языков программирования

- Состав: 1. Паспорт фонда оценочных средств
2. Описание оценочных средств:

Примерный перечень вопросов для самопроверки

Примеры вопросов для текущего контроля успеваемости (зачета)

Примеры вопросов для текущего контроля успеваемости (экзамена)

Составители:

Евтихов В. Г.

Москва, 2019 год

ПОКАЗАТЕЛЬ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Теория вычислительных процессов и языков программирования					
ФГОС ВО 09.03.01 «Информатика и вычислительная техника»					
В процессе освоения данной дисциплины студент формирует и демонстрирует следующие профессиональные и компетенции:					
КОМПЕТЕНЦИИ		Перечень компонентов	Технология формирования компетенций	Форма оценочного средства**	Степени уровней освоения компетенций
Индекс	Формулировка				
ПК-1	Обладать способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина»	<p>знать:</p> <ul style="list-style-type: none"> - методы и основные этапы трансляции; - основные положения теории вычислительных процессов и структур, их применение при реализации языков программирования и создании прикладных информационных систем; - принципы и способы технической реализации моделей процессов и структур; - теоретические основы синтаксического анализа, перевода, компиляции и машин Тьюринга. <p>уметь:</p> <ul style="list-style-type: none"> - определять различными способами формальные языки, корректно и свободно преобразовывать представления языка, применять теорию автоматов для решения практических задач; - строить конечные автоматы по заданным грамматикам, минимизировать автоматы, устранять левую рекурсию, приводить грамматику к нормальной форме Хомского, применять полученные знания для решения конкретных прикладных задач; - разрабатывать компоненты 	лекции, самостоятельная работа, лабораторная работа	УО	<p>пороговый уровень:</p> <p>знает теоретические основы синтаксического анализа, перевода, компиляции и машин Тьюринга, может применять теорию автоматов для решения практических задач, может применять основные положения теории вычислительных процессов и языков программирования при создании прикладных информационных систем;</p> <p>повышенный уровень:</p> <p>умеет строить конечные автоматы по заданным грамматикам, минимизировать автоматы, устранять левую рекурсию, приводить грамматику к нормальной форме Хомского, применять полученные знания для решения конкретных прикладных задач</p>

		<p>программных комплексов и баз данных, использовать современные инструментальные средства и технологии программирования.</p> <p>владеть:</p> <ul style="list-style-type: none"> - навыками использования методик теории формальных языков, автоматов и вычислений при решении практических задач; - основными навыками вычислительной теории и построения компиляторов; - навыками осваивать методики использования программных средств для решения практических задач. 			
--	--	--	--	--	--

** - Сокращения форм оценочных средств см. в приложении 2 к РП.

Перечень оценочных средств по дисциплине «Теория вычислительных процессов и языков программирования»

№ ОС	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Устный опрос собеседование, (УО)	Средство контроля, организованное как специальная беседа педагогического работника с обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п.	Вопросы по темам/разделам дисциплины
2	Зачет (З)	Средство промежуточной аттестации студента, проводится в письменной и/или устной форме.	Перечень вопросов по темам (разделам) дисциплины.
3	Экзамен (Э)	Средство промежуточной аттестации студента, проводится в письменно-устной форме.	Перечень вопросов по темам (разделам) дисциплины.

Примерный перечень вопросов для самопроверки по дисциплине «Теория вычислительных процессов и языков программирования» (ПК-1):

1. Перечислите элементы полного базиса и множества операторов класса стандартных схем программ.
2. Перечислите типы вершин графовой формы стандартной схемы и соответствующие им инструкции линейной формы.
3. Дайте определение понятия интерпретация стандартной схемы программы.
4. Запишите правила построения протокола выполнения программы.
5. Каким образом трактуют элементы базиса его свободные интерпретации?
6. Выполните подстановку для терма $f(x, y, z)[h(y)/x, a/y, g(h(x, z))/z]$.
7. Сравните «физические» интерпретации ОКА, МКА, ДГКА и ДДГКА.
8. Сформулируйте теоремы о неразрешимых свойствах стандартных схем программ.
9. Перечислите известные подходы к формализованному описанию семантики языков программирования.
10. Что является основной концепцией денотационной семантики?
11. Чем задается аксиоматическая семантика языка программирования?
12. Дайте определение инварианта контрольной точки программы. Приведите пример.
13. Сформулируйте законы консеквенции.
14. В чем разница между тестированием и верификацией программы?
15. Приведите примеры дискретных систем в ЭВМ.
16. Перечислите четыре основных черты дискретных систем.
17. Дайте формальное определение асинхронного процесса.
18. Назовите существующие частные виды асинхронных процессов.
19. Приведите пример асинхронного процесса, который является простым, но не управляемым; управляемым, но не простым.
20. Назовите этапы интерпретации асинхронного процесса.
21. Какая интерпретация асинхронного процесса называется диаграммой переходов?

22. Назовите классы параллельных вычислительных процессов.
23. Приведите примеры сотрудничающих и конкурирующих вычислительных процессов.
24. Приведите примеры SR и CR ресурсов в ЭВМ.
25. Перечислите известные аппаратные средства синхронизации взаимодействующих вычислительных процессов.
26. Назовите основной недостаток методов блокировки памяти и операции типа «проверка и установка».
27. Раскройте сущность семафорных механизмов Дейкстры.
28. Чем могут отличаться различные реализации семафорных механизмов?
29. Как можно обеспечить неделимость P- и V-операций в вычислительной системе?
30. Назовите основные достоинства почтового ящика как средства синхронизации взаимодействующих вычислительных процессов.
31. Запишите основные функции, которые используются для работы с почтовыми ящиками.
32. Что представляет собой функционирование конвейера?
33. Запишите основные системные запросы для работы с конвейером.
34. Назовите отличия работы очереди от конвейера.
35. Запишите основные функции управления работой очереди
36. Что называется последовательностью исполнения вычислительной схемы?
37. При каком условии процесс P_j системы будет заблокированным в состоянии S_l; находиться в тупике в состоянии S_l?
38. Какими элементами формально задается сеть Петри?
39. Какой переход сети Петри называется разрешенным?
40. Запишите правило срабатывания перехода сети Петри.
41. Какие условия должны выполняться для совместного срабатывания множества переходов?
42. Как построить дерево разметок сети Петри?
43. Запишите правило срабатывания множества совместно возможных переходов сети.
44. Какая позиция сети называется ограниченной, безопасной?

Примеры вопросов для текущего контроля успеваемости (зачета) по дисциплине «Теория вычислительных процессов и языков программирования» (ПК-1)

1. Приведите примеры 3-х местного терма и 4-х местного теста.
2. Проверьте схему S8 на тотальность и пустоту и постройте для нее функционально эквивалентную схему.
3. Для схемы S8 постройте цепочку и цепочку операторов.
4. Вычислите значение свободно интерпретированного терма $t = g(a, h(x, y))$, если терм-значение x есть 'f(f(x))', а терм-значение y есть 'g(x, a)'.
5. Задайте для схемы S8 свободную интерпретацию и постройте протокол выполнения полученной программы.
6. По схеме S8 определите термальное значение переменной y для пути, соответствующего цепочке схемы 0, 1, 20, 3, 1, 21, 4 и логико-термальную историю этого пути.
7. Изложите алгоритм преобразования ДГКА в ДДГКА.
8. Назовите этапы построения стандартной схемы, моделирующей заданный ДДГКА.

9. Для чего необходимо формальное представление семантики языка программирования?
10. Каким образом описывается операционная семантика операторов языков программирования?
11. Опишите вещественные числа с использованием денотационной семантики.
12. Запишите формальную спецификацию программы P_g на множестве состояний памяти State в аксиоматической семантике.
13. Как формально описываются семантические свойства отдельного оператора программы?
14. В чем смысл тройки Хоара? Приведите пример истинных троек Хоара для операторов присваивания, условия и цикла.
15. Опишите формально семантику типовых операторов какого-либо языка программирования.
16. Какими свойствами характеризуется корректность программы?
17. Запишите правила построений условий путей между соседними контрольными точками программы по методу математической индукции.
18. Сформулируйте этапы верификации программы на основе индуктивных утверждений.
19. Докажите полную корректность программы, вычисляющей наименьшее общее кратное двух целых неотрицательных чисел без использования деления и умножения.
20. Опишите семантику ограничивающей функции для циклов с верхним и нижним окончанием.
21. Изобразите и поясните типовую схему верификатора.
22. Какой асинхронный процесс называется эффективным, автономным?
23. В чем суть операции структурирования асинхронного процесса?
24. Приведите примеры модельной и предметной интерпретации асинхронного процесса.
25. Какая ситуация диаграммы переходов является конфликтной?
26. Приведите пример полумодулярной диаграммы переходов.
27. Опишите динамику функционирования маркированного графа.
28. Какая интерпретация маркированного графа называется сигнальным графом?
29. Сформулируйте классические задачи взаимодействия асинхронных процессов и назовите области их применения.
30. Какая ситуация при параллельном исполнении программы называется дедлоком? Приведите пример.
31. Перечислите условия, необходимые для возникновения тупика в вычислительной системе.
32. Охарактеризуйте классы ресурсов вычислительной системы.
33. Покажите, как можно решить проблему критического интервала с помощью операции типа «проверка и установка».
34. Проиллюстрируйте взаимное исключение n параллельных процессов с помощью семафорных механизмов.
35. Поясните механизм работы мониторов Хоара.
36. Каковы правила работы почтового ящика в зависимости от его сложности?
37. Назовите элементы вычислительной схемы асинхронного процесса?
38. Что определяют граф потока данных и граф управления вычислительной схемы?
39. Какая вычислительная схема называется детерминированной? Приведите пример.

40. Составьте вычислительную схему, содержащую условие соперничества операторов?
41. Какое состояние системы называется безопасным, тупиковым?
42. Опишите модель Холта (повторно используемых ресурсов).
43. В чем сущность алгоритма обнаружения тупика посредством редукции графа повторно используемых ресурсов?
44. Сформулируйте необходимое, достаточное, необходимо и достаточное условия существования тупика в модели Холта.
45. Изложите алгоритм обнаружения тупика по замкнутой цепочке запросов.
46. Как строится граф сети Петри?
47. Охарактеризуйте уровни активности переходов сети.
48. Сформулируйте постановку задачи достижимости и покрываемости сети Петри. Каким образом они разрешимы?
49. В чем разница между ограниченностью позиции сети первого и второго рода? Как их обнаружить в сети?
50. Какие типы вершин возможны на покрывающем дереве сети Петри?
51. Постройте сеть Петри, моделирующую схему алгоритма программы вычисления факториала числа.
52. Приведите пример сети Петри, моделирующей механизм взаимного исключения для трех параллельных процессов.

Примеры вопросов для текущего контроля успеваемости (экзамена) по дисциплине «Теория вычислительных процессов и языков программирования» (ПК-1)

1. Организация компилятора.
2. Множество цепочек. Операции над цепочками.
3. Структура компилятора.
4. Интерпретаторы, конверторы, кросс-компиляторы.
5. Определение языка. Синтаксис и семантика. Способы определения синтаксиса языка.
6. Лексический анализ.
7. Классификация грамматик. Иерархия Хомского.
8. Преобразование инфиксной записи в постфиксную для выражений.
9. Генерация кода.
10. Схемы программ. Основные понятия.
11. Виды взаимодействия вычислительных процессов. Понятие тупиковой ситуации.
12. Средства синхронизации вычислительных процессов.
13. Использование сетей Петри для моделирования взаимодействия вычислительных процессов.
14. Языки. Операции над языками. Итерация языка.
15. Алгоритмы. Частичные алгоритмы. Полные алгоритмы. Рекурсивные алгоритмы.
16. Синтаксис и семантика языков программирования.
17. Оптимизация кода. Исправление ошибок.
18. Способы определения языков. Грамматики. Грамматики с ограничениями на правила.
19. Распознаватели. Регулярные множества, их распознавание и порождение.
20. Регулярные множества и конечные автоматы. Графическое представление конечных автоматов.
21. Минимизация конечных автоматов. Алгоритм построения канонического конечного автомата.

22. Основные понятия контекстно-свободных грамматик. Преобразование КС-грамматик.
 23. Регулярные выражения и грамматики.
 24. Конечные автоматы.
 25. Двухуровневые и атрибутивные грамматики.
 26. Проблема разбора. Лево- и правосторонний разбор. Дерево разбора.
 27. Неоднозначность грамматик. Детерминированность разбора.
 28. Левая/правая рекурсия.
 29. Построение синтаксического графа.
 30. Построение программы синтаксического анализа для заданного синтаксиса.
 31. Восстановление при синтаксических ошибках.
 32. Работа с таблицей символов.
 33. Постфиксная запись. Вычисление постфиксных выражений.
 34. Нормальная форма Хомского. Нормальная форма Грейбах.
 35. Автоматы с магазинной памятью (МП-автоматы).
 36. Расширенные МП-автоматы.
 37. Недетерминированные и детерминированные МП-автоматы. Языки, допускаемые МП-автоматами.
 38. Эквивалентность МП-автоматов и КС-грамматик.
 39. Прямые и синтаксически ориентированные методы анализа.
 40. Нисходящая методика синтаксического анализа. LL(K)-грамматики.
- Алгоритмы

7. Учебно-методическое и информационное обеспечение дисциплины

Основная литература:

1. Дехтярь, М.И. Введение в схемы, автоматы и алгоритмы: Лекции по дискретной математике. [Электронный ресурс] – Электрон. дан. – М.: Национальный открытый университет «Интуит», 2016. – 169 с. – Режим доступа: <http://www.knigafund.ru/books/198017>. — Загл. с экрана.
2. Кузнецов, А.С., Царев, Р.Ю., Князьков, А.Н. Теория вычислительных процессов: Учебник. [Электронный ресурс] – Электрон. дан. – Красноярск: СФУ, 2015. – 118 с. – Режим доступа: <http://www.knigafund.ru/books/184651>. — Загл. с экрана.

Дополнительная литература:

1. Малявко, А.А. Формальные языки и компиляторы: Учебник. [Электронный ресурс] – Электрон. дан. – Новосибирск: НГТУ, 2014. – 431 с. – Режим доступа: <http://www.knigafund.ru/books/186311>. — Загл. с экрана.

в) программное обеспечение и интернет ресурсы

1. Операционная система Linux (свободное ПО)
 2. Офисные приложения LibreOffice для Linux (свободное ПО)
 3. Офисные приложения Microsoft Office 2013 (или ниже) - Microsoft Open License. Лицензия № 61984042
 4. Microsoft office 2013 prof (для обучения). Госконтракт № 18-09/14 от 22.09.2014 Акт № Тг 09950
 5. Visual Studio Professional w/MSDN ALNG LicSAPk OLP NL Academic Edition. Госконтракт № 18-09/14 от 22.09.2014 Акт № Тг064541 от 29.10.2014
- Электронные библиотеки университета (lib.mami.ru/lib/content/elektronyy-katalog;elib.mgup);
 - Электронно-библиотечная система «Издательства Лань» (www.e.lanbook.com);
 - Электронно-библиотечная система «Книгафонд» (www.knigafund.ru);
 - Научная электронная библиотека «КИБЕРЛЕНИНКА» (www.cyberleninka.ru);
 - Каталог образовательных Internet- ресурсов (<http://window.edu.ru>)

8. Материально-техническое обеспечение дисциплины

Учебно-вычислительные классы университета:

- ул. Автозаводская, 16. ауд. 1201. Аудитория общего фонда для лекционных, практических и семинарских занятий, оборудованная компьютерами, столами, стульями, аудиторной доской, проектором. Рабочее место преподавателя: компьютер, стол, стул;
- ул. Автозаводская, 16. ауд. 1202. Аудитория общего фонда для лекционных, практических и семинарских занятий, оборудованная компьютерами, столами, стульями, аудиторной доской, проектором. Рабочее место преподавателя: компьютер, стол, стул.

9. Методические рекомендации для самостоятельной работы студентов

Самостоятельная работа студентов по предмету «Теория вычислительных процессов и языков программирования» направления бакалавриата 09.03.01 «Информатика и вычислительная техника», профиль подготовки «Программное обеспечение средств вычислительной техники и автоматизированных систем» составляет 145 часов и подразделяется на следующие направления:

- работа с книгами и конспектом;
- работа с информационными источниками и методическими рекомендациями для подготовки к выполнению лабораторных работ.

Самостоятельная работа студента при обучении по заочной форме составляет основную долю учебного времени по данной дисциплине. Поэтому работа должна быть четко определена и структурирована.

Описание методики работы с книгами и конспектом:

- Ознакомьтесь с перечнем литературы по предмету;
- Изучите курс по темам, предварительно ознакомившись с содержанием каждой из них по программе;
- При первом чтении не задерживайтесь на математических выводах, а старайтесь получить общее представление об излагаемых вопросах, отмечая при этом трудные или неясные места;
- При повторном изучении темы усвойте все теоретические положения, математические зависимости и их выводы, а также принципы построения программ;
- Вникайте в сущность того или иного вопроса, а не пытайтесь запомнить отдельные факты и явления;
- Изучая курс, обращайтесь к предметному указателю в конце книги. Пока тот или иной раздел не усвоен, переходить к изучению новых разделов не следует. Краткий конспект курса будет полезен при повторении материала в период подготовки к зачету и защитам лабораторных работ;
- Чтобы лучше запомнить и усвоить изучаемый материал, необходимо иметь рабочую тетрадь и заносить в нее формулировки законов и основных понятий, новые незнакомые термины и названия, формулы, схемы, алгоритмы, языковые конструкции, функциональные зависимости и их выводы. Во всех случаях, когда материал поддается систематизации, составляйте графики, схемы, диаграммы, таблицы. Они облегчают изучение и уменьшают объем конспектируемого материала.

Описание методики работы с конспектом для подготовки к выполнению лабораторных работ:

К решению заданий можно приступить только тогда, когда будет усвоена определенная часть курса и разобраны типовые решения по соответствующей теме, приведенные в конспекте лекций. Решение заданий должно быть кратким, но четко обоснованным. При иллюстрации приведенного программного или математического решения заданий нужно приводить весь его ход и требуемые конструкции. При занесении результатов проведения самостоятельной работы в рабочую тетрадь все пункты должны быть аккуратно оформлены, текст должен быть четким и ясным. Данное требование позволит четче и организованнее подойти к зачету по дисциплине.

10. Методические рекомендации для преподавателя

10.1. План работы по дисциплине.

Преподавание дисциплины «Теория вычислительных процессов и языков программирования» осуществляется в соответствии с Федеральным государственным образовательным стандартом высшего профессионального образования по направлению подготовки 09.03.01 «Информатика и вычислительная техника», профиль подготовки «Программное обеспечение средств вычислительной техники и автоматизированных систем», квалификация - бакалавр.

Изучив глубоко содержание учебной дисциплины, целесообразно разработать план наиболее предпочтительных методов обучения и форм самостоятельной работы студентов, видам лекционных и лабораторных занятий.

10.2. Лекционное занятие

Лекционные занятия по дисциплине «Теория вычислительных процессов и языков программирования» проводятся согласно учебному плану в объеме 8 часов.

Вузовская лекция – одно из основных звеньев дидактического цикла обучения. Её целью является формирование у студентов ориентировочной основы для последующего усвоения материала методом самостоятельной работы. Содержание лекции должно отвечать следующим дидактическим требованиям:

- изложение материала от простого к сложному, от известного к неизвестному;

- логичность, четкость и ясность в изложении материала;
- возможность проблемного изложения, дискуссии, диалога с целью активизации деятельности студентов;
- опора смысловой части лекции на подлинные факты, события, явления, статистические данные;
- тесная связь теоретических положений и выводов с практикой и будущей профессиональной деятельностью студентов.

Преподаватель, читающий лекционные курсы в вузе, должен знать существующие в педагогической науке и используемые на практике варианты лекций, их дидактические и воспитывающие возможности, а также их методическое место в структуре процесса обучения.

При изложении материала важно помнить, что почти половина информации на лекции передается через интонацию. Учитывать тот факт, что первый кризис внимания студентов наступает на 15–20-й минутах, второй – на 30–35-й минутах. В профессиональном общении исходить из того, что восприятие лекций студентами младших и старших курсов существенно отличается по готовности и умению.

10.3. Лабораторные работы.

Лабораторные работы подготовлены в соответствии с:

- федеральным государственным образовательным стандартом высшего профессионального образования по направлению бакалавриата 09.03.01 «Информатика и вычислительная техника»;
- учебным планом по направлению 09.03.01 «Информатика и вычислительная техника», профиль подготовки «Программное обеспечение средств вычислительной техники и автоматизированных систем».

Лабораторные работы проводятся в объеме 27 часов, предусмотренном учебным планом по дисциплине «Теория вычислительных процессов и языков программирования».

Цель их проведения – развитие у студента умений и навыков в соответствии с компетенцией ПК-1, приведенных в п. 6.1.2. данной рабочей программы.

Задачи, стоящие перед студентами во время выполнения лабораторных работ:

- изучение информации по тематике работ;
- овладение навыками построения грамматик;
- овладение навыками работы с программными средствами при решении задач в профессиональной деятельности.

В ходе лабораторных занятий проводятся рассмотрение теоретического материала на практике. Каждое занятие состоит из двух частей: теоретической и практической. Теоретические знания, необходимые для практических занятий, даны в методических рекомендациях в виде перечня вопросов «для самопроверки» и вопросов, которые студенты могут извлечь из материала соответствующей лекции и путем самостоятельного изучения рекомендованной литературы. На занятиях преподаватель контролирует или совместно со студентами решает задачи, поставленные в работе.

Лабораторные работы проводятся с использованием компьютерной техники, по согласию методическим указаниям, предоставляемым студентам в электронном виде и перечнем индивидуальных заданий, выдаваемых преподавателем.

Тематика лабораторных работ разрабатывается преподавателем. Перечень лабораторных работ приведен в п.4.2. рабочей программы дисциплины «Теория вычислительных процессов и языков программирования», однако, по усмотрению руководителя задания могут быть изменены либо дополнены.

Порядок выполнения заданий включает в себя:

1. Оформление целей и задач практической работы.
2. Выполнение работы, в соответствии с заданием и предъявляемыми требованиями.

3. Оформление отчета в электронном (или печатном) виде.
4. Защиту работы.

Отчет о проделанных работах представляется студентом не менее, чем за сутки до защиты.

К защите работа представляется в сброшюрованном распечатанном или электронном виде и демонстрацией практической разработки. Последовательность брошюровки и оформление печатных работ и работ, приведенных в электронном виде, должны соответствовать требованиям по оформлению текстовой и графической документации.

Защита работ, проводится по результатам их выполнения, в виде демонстрации разработки и ответов на сопутствующие вопросы. Демонстрация работ осуществляется с использованием технических средств поддержки учебного процесса, включающих персональный компьютер.

Защита работ проводится по результатам их выполнения и является одним из факторов допуска к зачету и экзамену.

10.4. Самостоятельная работа.

При организации данного вида деятельности обучаемого необходимо предусмотреть развитие форм самостоятельной работы, выводя студентов к завершению изучения учебной дисциплины на её высший уровень.

Пакет вопросов для самостоятельной работы следует выдавать в начале семестра, определив предельные сроки их выполнения и сдачи. Задания для самостоятельной работы приведены в ФОС данной рабочей программы в виде вопросов для самопроверки.

Организуя самостоятельную работу, необходимо обучить студентов методам такой работы.

10.5. Проведение устного опроса студентов

Устный опрос позволяет контролировать процесс формирования знаний и умений студента, вместе с повторением и закреплением ранее изученного материала. При устном опросе принимаются диалоговые и монологические формы ответов, а также допускаются элементы дискуссионного общения. Устный опрос может проводиться при необходимости ведения оперативного наблюдения за реакцией студента для контроля уровня усвоения материала.

Устный опрос проводится в целях закрепления и проверки уровня усвоения изученного материала при подготовке к зачету.

10.6. Аттестация

При проведении аттестации студентов важно всегда помнить, что систематичность, объективность, аргументированность – главные принципы, на которых основаны контроль и оценка знаний студентов. Проверка, контроль и оценка знаний студента требуют учета его индивидуального стиля в осуществлении учебной деятельности. Знание критериев оценки знаний обязательно для преподавателя и студента.

10.6.1. Аттестация (зачет)

Согласно учебному плану в качестве аттестации студентов заочной формы обучения в 9 семестре предусмотрен зачет. Для облегчения подготовки к зачету, в ФОС данной рабочей программы приведен перечень вопросов для самоконтроля. Вопросы для самопроверки в совокупности с тематикой теоретической и практической частей дисциплины предназначены для подготовки студента к сдаче зачета. К зачету допускаются студенты, выполнившие и защитившие отчет по лабораторным работам и допущенные к аттестации. Зачет проводится в устной или письменной форме.

10.6.1. Аттестация (экзамен)

Согласно учебному плану в качестве аттестации студентов заочной формы

обучения в 9 семестре предусмотрен экзамен. Экзаменационная аттестация проводится по билетам (вопросам) в форме устного или письменного ответа студента. Для подготовки к ответу студенту выделяется не более 0,3 часа. Аттестация в форме экзамена проводится строго после аттестации в форме зачета. К экзамену допускается студент, выполнивший все предшествующие формы отчетности при наличии экзаменационной ведомости и зачетной книжки. ФОС по данной дисциплине содержит примерный перечень вопросов к экзамену. Следует отметить, что перечисленные выше вопросы могут входить в билеты с измененной формулировкой.

**Структура и содержание дисциплины «Теория вычислительных процессов и языков программирования»
по направлению подготовки 09.03.01 «Информатика и вычислительная техника»**

профиль подготовки: «Программное обеспечение средств вычислительной техники и автоматизированных систем». (Бакалавр)

n/n	Раздел	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов, и трудоемкость в часах					Виды самостоятельной работы студентов					Формы аттестации		
				Л	П/С	Лаб	СРС	КСР	К.Р.	К.П.	РГР	Реферат	К/р	Э	З	
1	Тема 1. Теория схем программ	9		1			20									
2	Тема 2. Семантическая теория программ	9		1			20									
3	Тема 3. Асинхронные процессы	9		1			20									
4	Тема 4. Взаимодействие процессов	9		2			20									
5	Тема 5. Модели вычислительных процессов	9		2			20									
6	Тема 6. Сети Петри	9		1			20									
7	Лабораторная работа 1. Алфавиты, языки и цепочки. Построение грамматик для различных, в том числе и контекстно-зависимых, языков	9				6	6									
8	Лабораторная работа 2. Написание регулярных выражений. Переход между регулярными выражениями, детерминированными конечными автоматами, недетерминированными конечными автоматами и праволинейными грамматиками. Минимизация конечных автоматов	9				6	6									
9	Лабораторная работа 3. Построение контекстно-свободных грамматик, удаление бесполезных и недостижимых символов. Устранение цепных и эpsilon-правил. Устранение левой рекурсии. Построение автоматов с магазинной памятью для КС-языков	9				7	6									
10	Лабораторная работа 4. Приведение	9				8	7									

	грамматик к нормальной форме Хомского. Построение машин Тьюринга. Вычислимость и невычислимость														
	Форма аттестации	9												+	+
	Всего часов по дисциплине			8		27	145								180