

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования «Кабардино-Балкарский государственный университет
им. Х.М. Бербекова» (КБГУ)

ИНСТИТУТ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА И ЦИФРОВЫХ ТЕХНОЛОГИЙ

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ИНФОРМАЦИОННОЙ
БЕЗОПАСНОСТИ

СОГЛАСОВАНО
Руководитель ОПОП



Т.Ю. Хаширова

«30» 05 2023 г.

УТВЕРЖДАЮ
Директор ИИИИИТ



А.Х. Шапсигов

«30» 05 2023 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.О.05.08 «ТЕОРИЯ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ И МЕТОДЫ ТРАНСЛЯЦИИ»

Направление подготовки

09.03.01 Информатика и вычислительная техника

Профиль

Программное обеспечение средств вычислительной техники и автоматизированных систем

Квалификация (степень) выпускника
бакалавр

Форма обучения
очная

Рабочая программа дисциплины «Теория языков программирования и методы трансляции» /сост. Г.А. Акбашева – Нальчик: ФГБОУ КБГУ, 2023. – 36 с.

Рабочая программа предназначена для преподавания дисциплины обязательной части студентам очной формы обучения по направлению подготовки 09.03.01 Информатика и вычислительная техника в 7 семестре 4 курса.

Рабочая программа дисциплины составлена с учетом Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.01 Информатика и вычислительная техника, утвержденного приказом Министерства образования и науки Российской Федерации от «19» сентября 2017 г. №929 (зарегистрировано в Минюсте России 10 октября 2017 г. № 48489).

Содержание

1. Цель и задачи освоения дисциплины	4
2. Место дисциплины в структуре ОПОП ВО	4
3. Требования к результатам освоения дисциплины.....	4
4. Содержание и структура дисциплины.....	5
5. Оценочные материалы для текущего и рубежного контроля успеваемости и промежуточной аттестации	10
6. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций	26
7. Учебно-методическое обеспечение дисциплины	27
8. Материально-техническое обеспечение дисциплины	33
9. Лист переутверждения рабочей программы дисциплины.....	35
Приложение.....	36

1. Цель и задачи освоения дисциплины

Цели освоения дисциплины:

Создание несложных языков программирования, используемых в системах автоматизации различных прикладных областей и разработка языковых процессоров для языков программирования различного уровня сложности, систематическое рассмотрение основных понятий языков программирования; синтаксиса, семантики, формальных способов описания языков программирования; типов данных, способов и механизмов управления данными; методов и основных этапов трансляции.

Задачи:

изучение методов разработки, описания и реализации языков программирования; формальных методов описания синтаксиса языка; методов синтаксического анализа современных языков программирования; формальных методов описания и реализации синтаксически управляемого перевода.

Изучение дисциплины направлено на подготовку специалистов, способных решать проблемы, возникающие при эксплуатации автоматизированных систем обработки информации и управления с учетом области, типов и задач профессиональной деятельности в соответствии с профессиональными стандартами:

- 06.001 – «Программист», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 18 ноября 2013 г. № 679н (зарегистрирован Министерством юстиции Российской Федерации 18 декабря 2013 г., регистрационный № 30635), с изменением, внесенным приказом Министерства труда и социальной защиты Российской Федерации от 12 декабря 2016 г. № 727н (зарегистрирован Министерством юстиции Российской Федерации 13 января 2017 г., регистрационный № 45230);
- 06.022 – «Системный аналитик», утвержденный приказом Минтруда России от 28.10.2014 № 809н (зарегистрирован в Минюсте России 24.11.2014 № 34882).

2. Место дисциплины в структуре ОПОП ВО

Дисциплина относится к дисциплинам обязательной части учебного плана по направлению 09.03.01 Информатика и вычислительная техника, профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем» Б1.О.05.09. Дисциплина является частью модуля «Информационные технологии и программирование».

Для изучения данной учебной дисциплины необходимы знания, умения и навыки, формируемые предшествующими дисциплинами: «Информатика», «Дискретная математика», «Языки и методы программирования», «Алгоритмические основы информатики», «Математическая логика и теория алгоритмов».

В результате освоения дисциплины обучающийся сможет частично продемонстрировать следующие обобщенные трудовые функции (ОТФ):

- Разработка требований и проектирование программного обеспечения (профессиональный стандарт 06.001 – «Программист», код D, уровень квалификации – 6).
- Концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности (профессиональный стандарт 06.022 – «Системный аналитик», код С, уровень квалификации – 6).

3. Требования к результатам освоения дисциплины

В совокупности с другими дисциплинами профиля «Программное обеспечение средств вычислительной техники и автоматизированных систем (ПОСВТиАС)» дисциплина «Теория языков программирования и методы трансляции» направлена на формирование следующих компетенций в соответствии с ФГОС ВО и ОПОП ВО по направлению

подготовки 09.03.01 – Информатика и вычислительная техника (ИиВТ) (уровень бакалавриата):

а) общепрофессиональные компетенции (ОПК):

- способен применять естественнонаучные и общетехнические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности (ОПК-1).

Коды и наименования индикаторов достижения компетенции:

ОПК-1.1. Знать основы высшей математики, физики, основы вычислительной техники и программирования.

ОПК-1.2. Уметь решать стандартные профессиональные задачи с применением естественнонаучных и общетехнических знаний, методов математического анализа и моделирования.

ОПК-1.3. Владеть навыками теоретического и экспериментального исследования объектов профессиональной деятельности.

В результате освоения дисциплины обучающийся должен:

Знать: основные модели, методы и алгоритмы теории языков программирования и методов трансляции, теоретические основы методов проектирования и способы описания языков программирования, основные положения теории формальных грамматик и языков, методов синтаксического анализа и перевода для класса формальных языков, используемых для описания основных конструкций языков программирования, стандарты, используемые для языков программирования;

Уметь: самостоятельно выполнять формальное описание синтаксиса и семантики, несложных процедурно-ориентированных и проблемно-ориентированных языков программирования, разрабатывать алгоритмы, реализующие методы синтаксического анализа и перевода для наиболее часто используемых классов формальных грамматик, пользоваться стандартными терминами и определениями, читать научные статьи и пользоваться литературой для самостоятельного решения научно-исследовательских задач, связанных с разработкой языков и реализацией систем программирования;

Иметь представление: о перспективных направлениях работ и методических подходах в области формальных методов описания и введения стандартов, используемых для описания языков программирования.

Владеть: методами разработки, описания и реализации языков программирования; формальными методами описания синтаксиса языка; методами синтаксического анализа современных языков программирования; формальными методами описания и реализации синтаксически управляемого перевода.

4. Содержание и структура дисциплины

В таблице 1 приводится описание содержания дисциплины, структурированное по разделам, с указанием по каждому разделу формы текущего контроля: защита лабораторной работы (ЛР), коллоквиум (К), рубежный контроль (РК), тестирование (Т).

Таблица 1

Содержание разделов дисциплины

№ раздела	Наименование раздела	Содержание раздела	Код контроля текущей компетенции	Форма текущего контроля
-----------	----------------------	--------------------	----------------------------------	-------------------------

			(или ее части)	
1	Основные концепции языков программирования	Основные концепции языков программирования. Парадигмы языков программирования. Критерии оценки языков программирования. Объекты данных в языках программирования. Механизмы типизации. Время жизни переменных. Основные концепции языков программирования. Область видимости переменных. Типы данных. Выражения и операторы присваивания. Подпрограммы.	ОПК-1	ТК, К, Т, ЛР
2	Описание языка программирования	Описание языка программирования. Определение синтаксиса языка: форма Бэкуса-Наура, диаграммы Вирта. Описание контекстных условий. Описание динамической семантики.	ОПК-1	ТК, К, Т, ЛР
3	Формальные грамматики и распознающие автоматы	Формальные грамматики и языки. Способы определения формальных языков. Классификация формальных грамматик. Выводы и деревья выводов. Неоднозначность грамматик. Непустые, конечные и бесконечные языки. Проблема принадлежности. Эквивалентные преобразования КС-грамматик. Нормальная форма Хомского. Нормальная форма Грейбах. Свойства замкнутости КС-языков. Конечные автоматы и преобразователи. Распознающий автомат. Конечный автомат. Способы задания конечных автоматов. Детерминированные конечные автоматы. Автоматные грамматики и конечные автоматы. Конечные преобразователи. Решение проблем принадлежности, пустоты, эквивалентности для конечных автоматов.	ОПК-1	ТК, К, Т, ЛР
4	Конечные автоматы и преобразователи	Конечные автоматы и преобразователи. Распознающий автомат. Конечный автомат. Способы задания конечных автоматов. Детерминированные конечные автоматы. Автоматные грамматики и конечные автоматы. Конечные преобразователи. Решение проблем принадлежности, пустоты,	ОПК-1	ТК, К, Т, ЛР

		эквивалентности для конечных автоматов.		
5	Автоматы и преобразователи с магазинной памятью	Автоматы и преобразователи с магазинной памятью. Определение автомата с магазинной памятью. Расширенные МП-автоматы. Эквивалентность МП-автоматов и КС-грамматик. Детерминированные МП-автоматы. Автоматы и преобразователи с магазинной памятью. Преобразователи с магазинной памятью.	ОПК-1	ТК, К, Т, ЛР
6	Общие методы синтаксического анализа	Общие методы синтаксического анализа. Определение разбора. Нисходящий разбор. Восходящий разбор. Моделирование недетерминированного МП-преобразователя. Алгоритм нисходящего разбора. Алгоритм восходящего разбора. Алгоритм Эрли.	ОПК-1	ТК, К, Т, ЛР
7	$LL(k)$ -грамматики	$LL(k)$ -грамматики. Определение $LL(k)$ -грамматик. Простые $LL(k)$ -грамматики. $LL(k)$ -грамматики. Алгоритм разбора для $LL(k)$ -грамматик. Рекурсивный спуск.	ОПК-1	ТК, К, Т, ЛР
8	$LR(k)$ -грамматики	$LR(k)$ -грамматики. Детерминированный разбор с помощью алгоритма «перенос-свертка». Алгоритм разбора для $LR(k)$ -грамматики. $LR(k)$ -грамматики. Построение $LR(k)$ -анализатора. Алгоритм построения анализатора для $LR(0)$ -грамматики без ε -правил. Алгоритм построения анализатора для $SLR(1)$ -грамматики без ε -правил. Включение ε -правил в $LR(0)$ - и $SLR(1)$ -грамматики.	ОПК-1	ТК, К, Т, ЛР
9	Грамматики предшествования	Грамматики предшествования. Понятие отношений предшествования. Алгоритм типа «перенос-свертка». Грамматики простого предшествования. Грамматики слабого предшествования. Грамматики операторного предшествования. Язык Флойда-Эванса.	ОПК-1	ТК, К, Т, ЛР
10	Промежуточные формы	Промежуточные формы представления программ. Польская	ОПК-1	ТК, К, Т, ЛР

	представления программ	запись: вычисление выражений, префиксная, постфиксная, инфиксная формы. Тетрады: переменные с индексами, указатели и функции, операторы. Триады. Байт-коды JVM.		
11	Формальные методы описания перевода	Формальные методы описания перевода. Перевод и семантика. СУ-схемы. Транслирующие грамматики. Атрибутные транслирующие грамматики: синтезированные атрибуты, унаследованные атрибуты, вычисление значений атрибутов.	ОПК-1	ТК, К, Т, ЛР
12	Разработка и реализация синтаксически управляемого перевода	Разработка и реализация синтаксически управляемого перевода. <i>L</i> -атрибутные и <i>S</i> -атрибутные транслирующие грамматики. Форма простого присваивания. Атрибутный перевод для <i>LL</i> (1)-грамматик. Реализация синтаксически управляемого перевода для транслирующей грамматики. <i>L</i> -атрибутный ДМП-процессор. Атрибутный перевод методом рекурсивного спуска. <i>S</i> -атрибутный ДМП-процессор. Математическая модель ДМП-процессора.	ОПК-1	ТК, К, Т, ЛР

Таблица 2

Структура дисциплины

Вид работы	Трудоемкость, часы
	7 семестр
Общая трудоемкость (в зачетных единицах)	4
Контактная работа (в часах):	56
Лекции (Л)	28
Практические занятия (ПЗ)	–
Семинарские занятия (СЗ)	–
Лабораторные работы (ЛР)	28
Самостоятельная работа (в часах):	61
Курсовой проект (КП), курсовая работа (КР)	–
Самостоятельное изучение разделов	61
Подготовка и прохождение промежуточной аттестации	27
Вид промежуточной аттестации	экзамен

Общая трудоемкость дисциплины составляет 4 зачетные единицы (144 часа).

Таблица 3

Лекции

№ п/п	Наименование разделов
-------	-----------------------

1	Основные концепции языков программирования
2	Описание языка программирования
3	Формальные грамматики и распознающие автоматы
4	Конечные автоматы и преобразователи
5	Автоматы и преобразователи с магазинной памятью
6	Общие методы синтаксического анализа
7	$LL(k)$ -грамматики
8	$LR(k)$ -грамматики
9	Грамматики предшествования
10	Промежуточные формы представления программ
11	Формальные методы описания перевода
12	Разработка и реализация синтаксически управляемого перевода
13	L-атрибутный ДМП-процессор. Атрибутный перевод методом рекурсивного спуска.
14	S-атрибутный ДМП-процессор. Математическая модель ДМП-процессора.

Таблица 4

Лабораторные работы

№ п/п	Наименование лабораторных работ
1.	Описание языка программирования – описание синтаксиса.
2.	Формальные грамматики и языки – эквивалентные преобразования КС-грамматик.
3.	Конечные автоматы и преобразователи – построение детерминированных конечных автоматов и преобразователей.
4.	Автоматы и преобразователи с магазинной памятью – построение МП-автоматов, расширенных МП-автоматов, ДМП-автоматов, ДМП-преобразователей.
5.	Общие методы синтаксического анализа – построение для заданной грамматики левого анализатора, правого анализатора, списков разбора.
6.	$LL(k)$ -грамматики – управляющая таблица для $LL(1)$ -грамматики, построение $LL(1)$ -грамматики, процедуры для метода рекурсивного спуска для $LL(1)$ -грамматик.
7.	$LR(k)$ -грамматики – управляющая таблица $LR(0)$ -анализатора для КС-грамматики, управляющая таблица $SLR(1)$ -анализатора для КС-грамматики.
8.	Грамматики предшествования – управляющая таблица для КС-грамматики слабого предшествования, управляющая таблица для КС-грамматики операторного предшествования, управляющая таблица для КС-грамматики простого предшествования.
9.	Промежуточные формы представления программ – ПОЛИЗ, тетрады и триады.
10	Формальные методы описания перевода

Практические занятия (семинары)

Не предусмотрено.

Курсовой проект (курсовая работа)

Не предусмотрено.

Таблица 5

Самостоятельное изучение разделов дисциплины

№ п/п	Вопросы, выносимые на самостоятельное изучение
1.	Подготовка к выполнению и защите лабораторных работ № 1-10. Написание отчетов по лабораторным работам.
2.	Основные концепции языков программирования

3.	Описание языка программирования: разработка описания синтаксиса основных конструкций языков программирования в форме Бэкуса-Наура (синтаксических диаграмм Вирта).
4.	Формальные грамматики и распознающие автоматы: построение дерева вывода, эквивалентные преобразования грамматик – примеры преобразования.
5.	Конечные автоматы и преобразователи: построение конечных автоматов и конечных преобразователей по грамматике – примеры.
6.	Автоматы и преобразователи с магазинной памятью: построение МП-автоматов и ДМП-автоматов – примеры.
7.	Общие методы синтаксического анализа: построение для грамматики левого и правого анализаторов – примеры.
8.	$LL(k)$ -грамматики: построение $LL(1)$ -грамматик, управляющих таблиц для $LL(1)$ -грамматик – примеры.
9.	$LR(k)$ -грамматики: построение $SLR(0)$ -анализатора для КС-грамматики – примеры.
10.	Грамматики предшествования: построение управляющей таблицы, моделирование анализатора типа «перенос-свертка» – примеры.
11.	Промежуточные формы представления программ: графическая и польская инверсная запись выражений – примеры.

5. Оценочные материалы для текущего и рубежного контроля успеваемости и промежуточной аттестации

Формы контроля текущих, рубежных и промежуточных знаний студентов по дисциплине определяются в соответствии с учебным планом образовательной программы и в соответствии с действующим Положением о балльно-рейтинговой системе оценки успеваемости студентов КБГУ.

От обучающихся требуется посещение занятий, выполнение лабораторных работ, знакомство с рекомендованной литературой.

При аттестации обучающихся оценивается качество работы на занятиях (умение вести дискуссию, способность четко и ёмко формулировать свои мысли), уровень подготовки к самостоятельной деятельности, качество выполнения заданий (презентаций, докладов, выполнение лабораторных работ и др.).

Конечными результатами освоения программы дисциплины являются сформированные когнитивные дескрипторы «знать», «уметь», «владеть», расписанные по отдельным компетенциям. Формирование этих дескрипторов происходит в течение всего семестра по этапам в рамках различного вида занятий и самостоятельной работы.

В ходе изучения дисциплины предусматриваются текущий, рубежный контроль и промежуточная аттестация.

5.1. Оценочные материалы для текущего контроля

Цель текущего контроля – оценка результатов работы в семестре и обеспечение своевременной обратной связи, для коррекции обучения, активизации самостоятельной работы обучающегося. Объектом текущего контроля являются конкретизированные результаты обучения (учебные достижения) по дисциплине.

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины «Теория языков программирования и методы трансляции» и включает: отчет по результатам выполнения лабораторных работ, самостоятельное выполнение индивидуальных домашних заданий с отчетом (защитой) в установленный срок.

Оценка качества подготовки на основании выполненных заданий ведется преподавателем (с обсуждением результатов), баллы начисляются в зависимости от сложности задания.

Критерии формирования оценок (оценивания) устного опроса

Устный опрос является одним из основных способов учёта знаний обучающегося по дисциплине. Развёрнутый ответ должен представлять собой связное, логически последовательное сообщение на заданную тему, показывать его умение применять определения.

В результате устного опроса знания, обучающегося оцениваются по следующей шкале:

Таблица 6

3 балла	2 балла	1 балл	0 баллов
ставится, если обучающийся: 1) полно излагает изученный материал, даёт правильное определенное экономических понятий; 2) обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные; 3) излагает материал последовательно и правильно с точки зрения норм литературного языка.	ставится, если обучающийся даёт ответ, удовлетворяющий тем же требованиям, что и для балла «1», но допускает 1-2 ошибки, которые сам же исправляет, и 1-2 недочёта в последовательности и языковом оформлении излагаемого.	ставится, если обучающийся обнаруживает знание и понимание основных положений данной темы, но: 1) излагает материал неполно и допускает неточности в определении понятий; 2) не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; 3) излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.	ставится, если обучающийся обнаруживает незнание большей части соответствующего раздела изучаемого материала, допускает ошибки в формулировке.

Баллы «1», «2», «3» могут ставиться не только за единовременный ответ, но и за рассредоточенный во времени, т.е. за сумму ответов, данных на протяжении занятия. начисляются в зависимости от сложности задания.

5.1.1. Оценочные материалы для самостоятельной работы обучающегося

Рабочая программа предусматривает проведение лекционных, лабораторных занятий, а также самостоятельную работу обучающихся. В ФГБОУ ВО «Кабардино-Балкарский государственный университет» действует балльно-рейтинговая система оценки учебных

достижений обучающихся по образовательным программам, реализуемым на основании федеральных государственных образовательных стандартов. Балльно-рейтинговая система оценки знаний является одной из составляющих системы управления качеством образовательной деятельности в университете.

Перечень типовых заданий для самостоятельной работы сформирован в соответствии с тематикой лабораторных занятий по дисциплине «Теория языков программирования и методы трансляции».

Темы для самостоятельной работы

1. Основные концепции языков программирования, анализ требований для некоторого языка программирования.
2. Описание языка программирования: разработка описания синтаксиса основных конструкций языков программирования в форме Бэкуса-Наура (синтаксических диаграмм Вирта) – примеры.
3. Формальные грамматики и распознающие автоматы: построение дерева вывода, эквивалентные преобразования грамматик – примеры преобразования.
4. Конечные автоматы и преобразователи: построение конечных автоматов и конечных преобразователей по грамматике – примеры.
5. Автоматы и преобразователи с магазинной памятью: построение МП-автоматов и ДМП-автоматов – примеры.
6. Общие методы синтаксического анализа: построение для грамматики левого и правого анализаторов – примеры.
7. $LL(k)$ -грамматики: построение $LL(1)$ -грамматик, управляющих таблиц для $LL(1)$ -грамматик – примеры.
8. $LR(k)$ -грамматики: построение $SLR(0)$ -анализатора для КС-грамматики – примеры.
9. Грамматики предшествования: построение управляющей таблицы, моделирование анализатора типа «перенос-свертка» – примеры.
10. Промежуточные формы представления программ: графическая и польская инверсная запись выражений – примеры.

Критерии формирования оценок по заданиям для самостоятельной работы студента

«Отлично» (5 баллов) – обучающийся показал глубокие знания материала по поставленным вопросам, грамотно, логично его излагает, структурировал и детализировал информацию, избегая простого повторения информации из текста, информация представлена в переработанном виде. Свободно использует необходимые формулы при решении задач;

«Хорошо» (4 балла) – обучающийся твердо знает материал, грамотно его излагает, не допускает существенных неточностей в процессе решения задач;

«Удовлетворительно» (3 балла) – обучающийся имеет знания основного материала по поставленным вопросам, но не усвоил его деталей, допускает отдельные неточности при решении задач;

«Неудовлетворительно» (менее 3 баллов) – обучающийся допускает грубые ошибки в ответе на поставленные вопросы и при решении задач

5.2. Оценочные материалы для рубежного контроля

Рубежный контроль осуществляется по более или менее самостоятельным разделам – учебным модулям курса и проводится по окончании изучения материала модуля в заранее установленное время. Рубежный контроль проводится с целью определения качества

усвоения материала учебного модуля в целом. В течение семестра проводится три таких контрольных мероприятия по графику.

В качестве форм рубежного контроля используется тестирование (письменное или компьютерное), проведение коллоквиума.

Рубежный контроль освоения студентом дисциплины осуществляется в рамках балльно-рейтинговой системы. Распределение баллов в соответствии с действующим Положением о балльно-рейтинговой системе оценки успеваемости студентов КБГУ приведено в таблице 7.

Таблица 7

Распределение баллов в соответствии с действующим Положением о балльно-рейтинговой системе

№ рейтинговой точки	Коллоквиум	Лаб.практикум	Посещаемость	Тестирование	Итого
1	7	8	3	5	23
2	7	8	3	5	23
3	7	8	4	5	24

Таблица 8

Критерии оценки

Вид мероприятия	Критерии оценки	Баллы
Коллоквиум (устный опрос по теме)	- ясность, четкость и доказательность изложения ответов на вопросы; - владение специальными терминами; - системность знаний по тематике	0-21 балл
Лабораторное занятие	- понимание цели и задач работы - выполнение заданий и обработка результатов - отчет и защита лабораторной работы	0-24 балла
Компьютерное тестирование по разделам дисциплины	Результаты тестирования (Количество баллов = $5 \cdot \varphi$, φ - доля правильно отвеченных тестов по теме).	0-15 баллов
Посещение занятий	При более 3 пропусках без уважительной причины занятий аннулируются баллы	0-10 баллов
Зачет	ясность, четкость и доказательность изложения ответов на вопросы; - владение специальными терминами; - системность знаний по тематике дисциплины в целом	0-30 баллов
Итоговая оценка		0-100 баллов

5.2.1. Оценочные материалы для проведения коллоквиума

Типовые задания первого коллоквиума (контролируемые компетенции ОПК-1):

Вариант №1

1. Какие языки называются императивными?
2. Чем подтип отличается от производного типа?
3. Какие существуют режимы и механизмы передачи параметров?
4. Определите понятия «синтаксис» и «семантика» языка.
5. Дайте определения цепочки и языка. Какие операции можно выполнять над цепочками символов и над языками?

6. Дайте определение распознавателя. Что представляют собой входная лента, устройство управления и вспомогательная память?
7. Опишите синтаксис оператора условного ветвления **switch** языка C. В качестве операторов, подлежащих исполнению при совпадении выражения с case-константой, можно использовать только операторы присваивания, правой частью которых являются арифметические выражения, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №2

1. Какие языки относят к языкам функционального программирования?
2. Из каких частей состоит среда ссылок?
3. Что представляет собой спецификация подпрограммы?
4. Что такое «метаязык»?
5. Каким образом можно классифицировать формальные грамматики?
6. Какое соответствие имеется между классами грамматик из иерархии Хомского и классами распознавателей, допускающих такой же класс языков?
7. Определите синтаксис условного оператора **if** языка Pascal, в котором условия задаются с помощью отношений, а в качестве операторов, выполняемых в зависимости от значений условий, можно использовать только операторы присваивания, правой частью которых являются арифметические выражения, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №3

1. Что понимается под объектом данных в языках программирования?
2. Какие существуют разновидности числовых типов, и чем определяется их многообразие?
3. Для каких целей используется таблица переходов при реализации многовариантных операторов ветвления?
4. Определите основную концепцию операционной семантики.
5. Определите связь между выводом и деревом вывода.
6. Дайте определение недетерминированного конечного автомата.
7. Задайте описание синтаксиса оператора цикла **repeat** языка Pascal. Условия в этом операторе задаются с помощью отношений, а в качестве операторов в теле цикла можно использовать только операторы присваивания. Правой частью операторов присваивания являются арифметические выражения, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №4

1. Что такое неявное определение типа?
2. Что содержит дескриптор вектора?
3. Перечислите основные достоинства и недостатки передачи параметров по ссылке.
4. Определите понятия «статическая семантика» и «динамическая семантика».
5. Дайте определение формального языка. Можно ли считать языки программирования формальными?
6. Дайте определение детерминированного конечного автомата.
7. Опишите синтаксис оператора варианта **case** языка Pascal. В качестве операторов, среди которых производится выбор для исполнения, используйте только операторы присваивания, правой частью которых являются арифметические выражения, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №5

1. Когда осуществляется статическое связывание типа?
2. Какими атрибутами характеризуются записи?
3. Перечислите основные достоинства и недостатки передачи параметров по значению, по результату, по значению и результату.
4. Опишите подход к определению семантики языка программирования с использованием аксиоматической семантики.
5. Что такое синтаксис и семантика языка? Какие средства существуют для описания синтаксиса и семантики?
6. Опишите алгоритм построения детерминированного конечного автомата, эквивалентного исходному недетерминированному конечному автомату.
7. Задайте синтаксис оператора цикла **do ... while** языка C. Условия в этом операторе представляют собой логические выражения, составленные из простых переменных булевского типа и логических операций И, ИЛИ, НЕ, а в качестве операторов в теле цикла можно использовать только операторы присваивания, правой частью которых являются логические выражения. Приоритет операций обычный.

Вариант №6

1. Что означает концептуальная целостность языка программирования?
2. Какие существуют способы представления множеств в памяти?
3. Что означают сужающее и расширяющее преобразования типа?
4. Чем отличается операционная семантика от денотационной?
5. Дайте определение формальной грамматики.
6. Опишите алгоритм построения недетерминированного конечного автомата по автоматной грамматике.
7. Опишите синтаксис оператора цикла **while** некоторого подмножества языка C. Условия в этом операторе задаются с помощью отношений, а в качестве операторов в теле цикла можно использовать только операторы присваивания, правой частью которых являются арифметические выражения, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №7

1. Что определяет область видимости переменных?
2. Как выполняется сокращенное вычисление?
3. Чем сопрограммы отличаются от обычных подпрограмм?
4. Опишите основную концепцию W-грамматик и расскажите, как они используются для задания синтаксиса и семантики языков программирования?
5. Дайте определение отношения выводимости. Какие виды выводов наиболее часто используются?
6. Дайте определение распознавателя. Что представляют собой входная лента, устройство управления и вспомогательная память?
7. Разработайте описание синтаксиса оператора цикла **for** языка C. В качестве операторов в теле цикла можно использовать только операторы присваивания, правой частью которых являются арифметические выражения целого типа, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №8

1. Что понимается под динамической областью видимости имен?
2. Что такое перегруженный оператор?
3. Какие языки являются декларативными?
4. Определите наиболее характерные типы контекстных условий.
5. Перечислите и приведите примеры различных способов определения бесконечных языков.

6. Какое соответствие имеется между классами грамматик из иерархии Хомского и классами распознавателей, допускающих такой же класс языков?
7. Определите синтаксис условного оператора **if** языка C, в котором условия задаются с помощью отношений, а в качестве операторов, выполняемых в зависимости от значений условий, можно использовать только операторы присваивания. Правой частью операторов присваивания являются арифметические выражения, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №9

1. Каким образом реализуется статическая область видимости имен?
2. Какие типы побочных эффектов встречаются в выражениях?
3. Перечислите основные достоинства и недостатки передачи параметров по имени.
4. Какие расширения БНФ используются наиболее часто?
5. Неоднозначность грамматик.
6. Дайте определение недетерминированного конечного автомата.
7. Задайте описание синтаксиса оператора цикла **repeat** языка Pascal. Условия в этом операторе задаются с помощью отношений, а в качестве операторов в теле цикла можно использовать только операторы присваивания. Правой частью операторов присваивания являются арифметические выражения, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №10

1. Чем отличается литерал от именованной константы?
2. Назовите недостатки слабой типизации языка программирования.
3. Как определяется эквивалентность типов?
4. Определите основную концепцию денотационной семантики.
5. Бесконечные языки.
6. Дайте определение детерминированного конечного автомата.
7. Определите синтаксис условного оператора **if** языка Pascal, в котором условия задаются с помощью отношений, а в качестве операторов, выполняемых в зависимости от значений условий, можно использовать только операторы присваивания, правой частью которых являются арифметические выражения, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №11

1. С помощью каких атрибутов можно охарактеризовать переменную?
2. Назовите основные признаки строгой типизации.
3. Каким образом разрешаются коллизии при использовании хеширования?
4. Форма Бэкуса-Наура.
5. Нормальная форма Хомского.
6. Опишите алгоритм построения детерминированного конечного автомата, эквивалентного исходному недетерминированному конечному автомату.
7. Разработайте описание синтаксиса оператора цикла **for** языка C. В качестве операторов в теле цикла можно использовать только операторы присваивания, правой частью которых являются арифметические выражения целого типа, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №12

1. Что означает предопределенное имя?

2. Каким образом производный тип наследует атрибуты?
3. Чем различаются свободные и размеченные объединения?
4. Синтаксические диаграммы Вирта.
5. Нормальная форма Грейбах.
6. Опишите алгоритм построения недетерминированного конечного автомата по автоматной грамматике.
7. Опишите синтаксис оператора условного ветвления **switch** языка C. В качестве операторов, подлежащих исполнению при совпадении выражения с case-константой, можно использовать только операторы присваивания, правой частью которых являются арифметические выражения, составленные из простых переменных и констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №13

1. Из каких составляющих складывается суммарная стоимость языка программирования?
2. Когда осуществляется статическое связывание типа?
3. Что представляет собой составной оператор присваивания?
4. Описание контекстных условий в языках программирования.
5. Свойство замкнутости КС-языков.
6. Дайте определение распознавателя. Что представляют собой входная лента, устройство управления и вспомогательная память?
7. Разработайте описание синтаксиса условного оператора **if** языка Pascal. Условия в этом операторе задаются логическими выражениями, составленными из простых переменных булевского типа и логических операций И, ИЛИ, НЕ, а в качестве операторов можно использовать только операторы присваивания, правой частью которых являются логические выражения. Приоритет операций обычный.

Вариант №14

1. Чем определяется порядок вычисления операций в арифметических выражениях?
2. Какие проблемы возникают при разработке операторов цикла с параметром?
3. Какие операции определены для указателей?
4. Денотационная семантика.
5. Способы определения формальных языков.
6. Опишите алгоритм построения детерминированного конечного автомата, эквивалентного исходному недетерминированному конечному автомату.
7. Опишите синтаксис оператора цикла **while** некоторого подмножества языка Pascal. Условия в этом операторе задаются с помощью отношений, а в качестве операторов в теле цикла можно использовать только операторы присваивания, правой частью которых являются арифметические выражения, составленные из простых переменных и констант вещественного типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Вариант №15

1. В каких случаях и почему более предпочтительным, чем оператор **case**, является многовариантный оператор ветвления **elsif**?
2. Какие проблемы возникают при разработке операторов цикла с логическим управлением?
3. Что содержит дескриптор вектора?
4. Исключение рекурсии из БНФ.
5. Классификация формальных языков.
6. Опишите алгоритм построения недетерминированного конечного автомата по автоматной грамматике.
7. Опишите синтаксис условного оператора присваивания языка C. В правой части такого оператора для задания условий следует использовать отношения и арифметические выражения целого типа, составленные из простых переменных и

констант целого типа, круглых скобок и знаков арифметических операций: сложения, вычитания, умножения и деления. Приоритет операций обычный.

Типовые задания второго коллоквиума (контролируемые компетенции ОПК-1):

Вариант №1

1. Определите такие понятия, как конфигурация МП-автомата, такт работы МП-автомата, ε -такт.
2. Опишите алгоритм, позволяющий по заданному ДМП-автомату построить эквивалентный ему дочитывающий ДМП-автомат.
3. Как используются списки разбора в алгоритме Эрли?
4. Какие КС-грамматики называются s -грамматиками?
5. Как строится магазинный алфавит для алгоритма типа «перенос-свертка»?

Вариант №2

1. Дайте определение недетерминированного расширенного автомата с магазинной памятью. Что представляют собой входная лента, устройство управления и вспомогательная память расширенного МП-автомата?
2. Опишите алгоритм построения множества зацикливающих конфигураций.
3. Как выполняется возврат в алгоритме восходящего разбора с возвратами?
4. Что такое множества выбора и как они используются при анализе s -грамматик?
5. Дайте определение $LR(k)$ -грамматики. В чем особенности этого класса грамматик?

Вариант №3

1. Определите такие понятия, как конфигурация расширенного МП-автомата, такт работы МП-автомата, ε -такт.
2. Будет ли МП-преобразователь детерминированным, если он построен на основе ДМП-автомата? Ответ пояснить.
3. Для чего используются магазины L_1 и L_2 в алгоритме нисходящего разбора?
4. Дайте определения s -, q - и $LL(1)$ -грамматик. Как соотносятся классы этих грамматик?
5. Что такое «грамматическое вхождение»?

Вариант №4

1. Опишите процесс построения МП-автомата по расширенному МП-автомату.
2. Дайте определение детерминированного расширенного МП-автомата.
3. Какой МП-автомат называется незацикливающимся?
4. Как определяются множества $FIRST_k(\beta)$ и $FOLLOW_k(A)$?
5. Перечислите основные функции алгоритма $LR(k)$ -алгоритма разбора. Как для него определяются функции действия и перехода?

Вариант №5

1. Дайте определение МП-автомата, допускающего входные цепочки опустошением магазина.
2. Дайте определение недетерминированного МП-преобразователя и недетерминированного расширенного МП-преобразователя.
3. Дайте определение основы. Какие цепочки могут не иметь основы?
4. Что представляет собой головной модуль в синтаксическом анализаторе, реализующем метод рекурсивного спуска?
5. Как строится конечный автомат для распознавания активных префиксов? Этот автомат детерминированный или нет?

Вариант №6

1. Опишите процесс построения МП-автомата, допускающего входные цепочки опустошением магазина.
2. Дайте определение детерминированного МП-преобразователя и детерминированного расширенного МП-преобразователя.
3. Какие основные действия выполняет правый анализатор?

4. Какие требования предъявляются к языку программирования для реализации метода рекурсивного спуска?
5. Дайте определения множеств *OFIRST* и *OBLOW*. Каким образом они могут быть вычислены?

Вариант №7

1. Опишите алгоритм построения недетерминированного МП-автомата по КС-грамматике.
2. Дайте определение дочитывающего ДМП-автомата.
3. Какова процедура построения дерева разбора при восходящем анализе?
4. Как программируются процедуры для распознавания нетерминалов в методе рекурсивного спуска?
5. Какие изменения нужно внести в алгоритм построения *SLR(1)*-анализатора для учета правил с пустой правой частью?

Вариант №8

1. Опишите алгоритм построения недетерминированного расширенного МП-автомата по КС-грамматике.
2. Приведите лемму о существовании ДМП-автомата, для которого всегда возможен очередной такт.
3. Какие основные действия выполняет левый анализатор?
4. Какие грамматики называются *LL(k)*-грамматиками?
5. Перечислите основные функции алгоритма *LR(k)*-алгоритма разбора. Как для него определяются функции действия и перехода?

Вариант №9

1. Опишите алгоритм построения КС-грамматики по недетерминированному МП-автомату.
2. Дайте определение перевода, определяемого МП-преобразователем.
3. Какова процедура построения дерева разбора при нисходящем анализе?
4. Как описывается такт работы *LL(1)*-анализатора?
5. Как строится магазинный алфавит для алгоритма типа «перенос-свертка»?

Вариант №10

1. Дайте определение недетерминированного автомата с магазинной памятью и языка, допускаемого МП-автоматом.
2. Дайте определение детерминированного МП-автомата. Дайте определение заикливающей конфигурации.
3. Дайте определение левого и правого разборов. Как левый и правый разборы связаны с левым и правым выводом?
4. Дайте определения *s*-, *q*- и *LL(1)*-грамматик. Как соотносятся классы этих грамматик?
5. Как строится конечный автомат для распознавания активных префиксов? Этот автомат детерминированный или нет?

Типовые задания третьего коолоквиума (контролируемые компетенции ОПК-1):

Вариант №1

1. Дайте определение алгоритма типа «перенос-свертка». Каким образом описывается его работа?
2. Почему исходная программа переводится сначала в промежуточное представление, а затем в объектный код?
3. Как определяется понятие семантики языков программирования?
4. Что такое входная и выходная грамматики транслирующей грамматики?
5. Как программируются процедуры в методе рекурсивного спуска, реализующего перевод, описываемый транслирующей грамматикой?

Вариант №2

1. Дайте определение грамматик простого и слабого предшествования. Чем различаются эти грамматики?

2. Как арифметическое выражение записывается в ПОЛИЗ? Как в ПОЛИЗ записывается унарный минус?
3. Как можно описать простейшие переводы?
4. Как связаны вывод цепочки и дерево вывода активной цепочки?
5. Опишите, каким образом представляются в магазине и вычисляются унаследованные и синтезированные атрибуты символов грамматики в S-атрибутном ДМП-процессоре.

Вариант №3

1. Как определяются отношения предшествования? Какие средства существуют для их определения?
2. Почему в компиляторах используется для представления выражении польская инверсная запись?
3. Дайте определение СУ-схемы. Какие цепочки называются входными и выходными?
4. В чем основные различия между унаследованными и синтезированными атрибутами?
5. Как программируются процедуры в методе рекурсивного спуска, реализующего перевод, описываемый L-атрибутной транслирующей грамматикой?

Вариант №4

1. Как представляются отношения предшествования? Как они используются при выполнении синтаксического анализа?
2. Как производится вычисление выражения, представленного в польской инверсной записи?
3. Какие переводы называются синтаксически управляемыми? Что такое элемент перевода?
4. Какая грамматика называется атрибутной транслирующей грамматикой?
5. С какой целью осуществляется переименование имен атрибутов нетерминальных символов из левых частей правил вывода L-атрибутной транслирующей грамматики при реализации вывода L-атрибутного перевода методом рекурсивного спуска?

Вариант №5

1. Язык Флойда-Эванса. Для чего может быть использован язык Флойда-Эванса?
2. Как графически представляется польская инверсная запись?
3. Как происходит порождение пар цепочек под управлением СУ-схемы?
4. Что представляет собой правило вычисления атрибутов?
5. Приведите процедуру преобразования ДМП-процессора в L-атрибутный ДМП-процессор.

Вариант №6

1. Как можно представить в тетрадах переменную с индексами, указатель функции?
2. Дайте определение алгоритма типа «перенос-свертка». Каким образом описывается его работа?
3. Какие СУ-схемы называются простыми?
4. Как граф зависимостей используется при вычислении атрибутов?
5. Опишите, каким образом представляются в магазине и вычисляются унаследованные и синтезированные атрибуты символов грамматики в L-атрибутном ДМП-процессоре.

Вариант №7

1. Какие переводы позволяют описать простые СУ-схемы?
2. Чем различаются триады и косвенные триады?
3. Дайте определение грамматик простого и слабого предшествования. Чем различаются эти грамматики?
4. Из каких этапов состоит процесс построения транслирующей грамматики?
5. Приведите алгоритм построения управляющей таблицы для транслирующей грамматики цепочечного перевода, входной грамматикой которой является $LL(1)$ -грамматика.

Вариант №8

1. Что представляет собой виртуальная машина Java?

2. Как определяются отношения предшествования? Какие средства существуют для их определения?
3. Как определить входную и выходную грамматики СУ-схемы?
4. Какие типы ошибок позволяет найти тестирование АТ-грамматики?
5. В каком случае АТ-грамматика имеет форму простого присваивания? Приведите процедуру преобразования произвольной АТ-грамматики в форму простого присваивания.

Вариант №9

1. Как представляются отношения предшествования? Как они используются при выполнении синтаксического анализа?
2. Как обрабатываются инструкции байт-кода?
3. Как происходит преобразование деревьев под управлением СУ-схем?
4. Из каких этапов состоит процесс построения АТ-грамматики?
5. Какое правило вычисления атрибутов называется копирующим?

Вариант №10

1. Какая грамматика называется транслирующей грамматикой?
2. Постфиксная, префиксная, инфиксная формы записи выражения.
3. Язык Флойда-Эванса. Для чего может быть использован язык Флойда-Эванса?
4. Как определяется активная цепочка, ее входная и операционная части?
5. Дайте определение L-атрибутной и S-атрибутной транслирующих грамматик.

Критерии формирования оценок по контрольным точкам (коллоквиум)

4 балла ставится за работу, выполненную полностью без ошибок и недочетов; обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, решено 100% задач;

3 балла ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов. Обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, допуская незначительные неточности при решении задач, решено 70% задач;

2 балла ставится за работу, если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой. Обучающийся затрудняется с правильной оценкой предложенной задачи, дает неполный ответ, решено 55% задач

1 балл ставится за работу, если число ошибок и недочетов превысило норму или правильно выполнено менее 2/3 всей работы. Обучающийся дает неверную оценку ситуации, решено менее 50 % задач.

5.2.2. Оценочные материалы: тестирование

Тестирование обучающихся проводится в онлайн-режиме согласно расписанию в ЭИОС open.kbsu.ru

Примерные тестовые задания (контролируемые компетенции ОПК-1)

Полный перечень тестовых заданий представлен в ЭОИС

I:

S: Строка символов, используемая для обозначения некоторой сущности в программе это – ###.

+: идентификатор

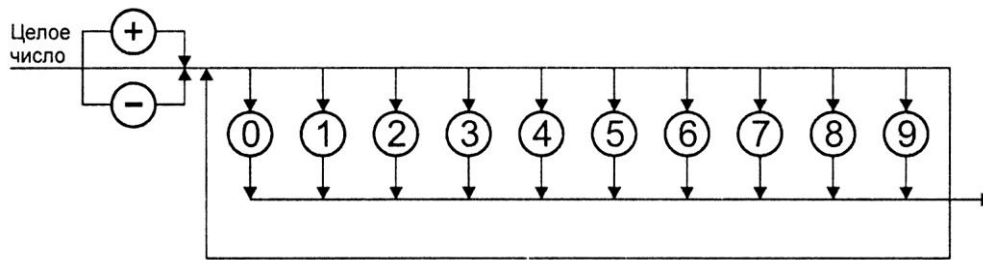
I:

- S: Идентификаторы, имеющие значение только в определенном контексте это – ###.
- + : ключевые слова
- I:
- S: Идентификаторы, имеющие конкретный смысл, но не являющиеся ключевыми словами это – ### имена.
- + : предопределенные
- I:
- S: Объект данных, имя которого связано со значением в течение всего времени жизни это – ###.
- + : константа
- I:
- S: Объект данных, который явным образом определен и именован в программе это – ###.
- + : переменная
- I:
- S: Статическое определение типа переменной осуществляется:
- + : При трансляции программы
 - : Во время выполнения программы
 - : В зависимости от настроек компилятора
- I:
- S: Динамическое определение типа переменной осуществляется:
- : При трансляции программы
 - + : Во время выполнения программы
 - : В зависимости от настроек компилятора
- I:
- S: Языки программирования, в которых информация о типе данных используется только для обеспечения корректности программ на машинном уровне это языки:
- + : Со слабой типизацией
 - : Со строгой типизацией
 - : И со слабой типизацией и со строгой типизацией
- I:
- S: Типизация, обеспечивающая полный контроль типов (статический и динамический) это:
- : Слабая типизация
 - + : Строгая типизация
 - : Динамическая типизация
 - : Статическая типизация
- I:
- S: Два объекта принадлежат эквивалентным типам, если у них одинаковая структура при:
- + : Структурной эквивалентности
 - : Именной эквивалентности
 - : Структурной и именной эквивалентности
- I:
- S: Два объекта принадлежат эквивалентным типам, если они описаны с помощью одного и того же имени типа при:
- : Структурной эквивалентности
 - + : Именной эквивалентности

-: Структурной и именной эквивалентности

I:

S: На рисунке приведены



+: Синтаксические диаграммы Вирта для определения множества целых чисел со знаком

-: Синтаксические диаграммы Вирта для определения множества вещественных чисел со знаком

-: Множество десятичных чисел с фиксированной точкой

-: Синтаксические диаграммы Вирта для языка арифметических выражений с четырьмя основными операциями

-: Синтаксические диаграммы Вирта для множества арифметических правил

I:

S: Формальная аппликативная модель для описания семантики языков программирования, это:

-: Грамматические модели

-: Операционная семантика

-: Аксиоматическая семантика

+: Денотационная семантика

I:

S: БНФ $\langle \text{одн} \rangle ::= 0 | 1 | \langle \text{одн} \rangle 0 | \langle \text{одн} \rangle 1$ описывает

+: Синтаксис двоичных чисел

-: Синтаксис десятичных чисел

-: Операционный метод описания семантики

-: Денотационную семантику

I:

S: Статическое определение типа переменной осуществляется:

+: При трансляции программы

-: Во время выполнения программы

-: В зависимости от настроек компилятора

I:

S: Динамическое определение типа переменной осуществляется:

-: При трансляции программы

+: Во время выполнения программы

-: В зависимости от настроек компилятора

I:

S: Языки программирования, в которых информация о типе данных используется только для обеспечения корректности программ на машинном уровне это языки:

+: Со слабой типизацией

-: Со строгой типизацией

-: И со слабой типизацией и со строгой типизацией

Критерии формирования оценок по тестовым заданиям

Выполнение тестирования оценивается согласно проценту правильных ответов. Максимально возможное количество баллов за тестирование – 5.

5.3. Оценочные материалы для промежуточной аттестации

Целью промежуточных аттестаций по дисциплине является оценка качества освоения дисциплины обучающимися.

Промежуточная аттестация предназначена для объективного подтверждения и оценивания достигнутых результатов обучения после завершения изучения дисциплины. Осуществляется в конце семестра и представляет собой итоговую оценку знаний по дисциплине «Теория языков программирования и методы трансляции» в виде проведения экзамена.

Промежуточная аттестация может проводиться в устной, письменной форме, и в форме тестирования. На промежуточную аттестацию отводится до 30 баллов.

Перечень вопросов для подготовки к экзамену (контролируемые компетенции ОПК-1)

1. Парадигмы языков программирования: императивные языки, языки функционального программирования, декларативные языки, объектно-ориентированные языки
2. Критерии оценки языков программирования: понятность языка программирования, надежность языка программирования, гибкость языка программирования, простота языка программирования, естественность языка программирования, мобильность языка программирования, стоимость языка программирования
3. Объекты данных в языках программирования
4. Механизмы типизации: статические и динамические типы данных, слабая типизация, строгая типизация, производные типы, эквивалентность типов, наследование атрибутов, подтипы, анонимные типы и подтипы
5. Время жизни переменных. Область видимости переменных.
6. Элементарные типы данных. Числовые типы, логический тип, символьный тип, указатели, символьные строки, перечислимые типы, ограниченные типы.
7. Векторы и массивы: спецификация и реализация в языках программирования.
8. Записи и массивы со структурированными компонентами: спецификация и реализация в языках программирования.
9. Объединения, множества, списки: спецификация и реализация в языках программирования.
10. Выражения и операторы присваивания: спецификация и реализация в языках программирования.
11. Структуры управления на уровне операторов: спецификация и реализация в языках программирования.
12. Составные операторы, условные операторы, операторы if, операторы многовариантного ветвления: спецификация и реализация в языках программирования.
13. Операторы цикла: простая итерация, циклы с параметром, цикл с предусловием, цикл с постусловием, циклы, управляемые пользователем: спецификация и реализация в языках программирования.
14. Подпрограммы. Определение подпрограммы. Формальные и фактические параметры подпрограммы.
15. Процедуры и функции. Методы передачи параметров. Сопрограммы.

16. Определение синтаксиса языка. Форма Бэкуса-Наура. Синтаксические диаграммы Вирта.
17. Описание контекстных условий. Описание динамической семантики. Грамматические модели. Операционная семантика. Аксиоматическая семантика. Денотационная семантика.
18. Формальные грамматики и языки. Способы определения формальных языков.
19. Формальные грамматики. Классификация формальных грамматик.
20. Формальные грамматики. Выводы и деревья выводов. Неоднозначность грамматик.
21. Непустые, конечные и бесконечные языки. Проблема принадлежности
22. Эквивалентные преобразования КС-грамматик.
23. Нормальная форма Хомского.
24. Нормальная форма Грейбах.
25. Свойства замкнутости КС-языков.
26. Конечные автоматы и преобразователи.
27. Распознающий автомат. Конечный автомат.
28. Способы задания конечных автоматов.
29. Детерминированные конечные автоматы.
30. Автоматные грамматики и конечные автоматы.
31. Решение проблемы принадлежности для конечных автоматов.
32. Решение проблемы пустоты языка для конечных автоматов.
33. Решение проблемы эквивалентности для конечных автоматов.
34. Конечные преобразователи.
35. Автоматы и преобразователи с магазинной памятью.
36. Расширенные МП-автоматы.
37. Эквивалентность МП-автоматов и КС-грамматик.
38. Детерминированные МП-автоматы.
39. Преобразователи с магазинной памятью.
40. Общие методы синтаксического анализа.
41. Определение разбора. Нисходящий разбор. Восходящий разбор.
42. Моделирование недетерминированного МП-преобразователя.
43. Алгоритм нисходящего разбора.
44. Алгоритм восходящего разбора.
45. Алгоритм Эрли.
46. LL(k)-грамматики. Простые LL(1)-грамматики. Определение LL(1)-грамматики. Алгоритм разбора для LL(1)-грамматик. Рекурсивный спуск.
47. LR(k)-грамматики. Детерминированный разбор с помощью алгоритма "перенос-свертка". Алгоритм разбора для LR(k)-грамматики. Построение LR(k)-анализаторов.
48. Алгоритм построения анализатора для LR(0)-грамматики без ϵ -правил. Алгоритм построения анализатора для LR(1)-грамматики без ϵ -правил. Включение ϵ -правил в LR(0)- и SLR(1)-грамматики.
49. Грамматики предшествования. Алгоритм типа «перенос-свертка». Грамматики простого предшествования. Грамматики слабого предшествования. Грамматики операторного предшествования.
50. Язык Флойда-Эванса.
51. Промежуточные формы представления программ. Польская запись. Вычисление выражений. Префиксная форма. Постфиксная форма. Свойства ПОЛИЗ. Графическое представление выражений.
52. ПОЛИЗ как промежуточный язык. Элементы массивов. Указатели функций. Условные выражения. Оператор присваивания. Условный оператор. Оператор цикла.
53. Тетрады. Переменные с индексами. Указатели функций. Операторы.
54. Триады. Переменные с индексами. Указатели функций. Операторы.
55. Байт-коды JVM.

56. Формальные методы описания перевода. Перевод и семантика. СУ-схемы.
57. Транслирующие грамматики.
58. Атрибутные транслирующие грамматики.
59. Синтезированные атрибуты.
60. Определение атрибутной транслирующей грамматики. Вычисление значений атрибутов.
61. Методика разработки описания перевода. Пример разработки АТ-грамматики.
62. Разработка и реализация синтаксически управляемого перевода.
63. L-атрибутные и S-атрибутные транслирующие грамматики.
64. Форма простого присваивания.
65. Атрибутный перевод для LL(1)-грамматик.
66. Реализация синтаксически управляемого перевода для транслирующей грамматики.
67. L-атрибутный ДМП-процессор.
68. Атрибутный перевод методом рекурсивного спуска.
69. S-атрибутный ДМП-процессор.
70. Математическая модель восходящего ДМП-процессора.
71. Реализация S-атрибутного ДМП-процессора.

Критерии формирования оценок по промежуточной аттестации

«Отлично» получают обучающиеся, которые свободно ориентируются в материале и отвечают без затруднений. Обучающийся способен к выполнению сложных заданий, постановке целей и выборе путей их реализации. Работа выполнена полностью без ошибок, решено 100% задач;

«Хорошо» получают обучающиеся, которые относительно полно ориентируются в материале, отвечают без затруднений, допускают незначительное количество ошибок. Обучающийся способен к выполнению сложных заданий. Работа выполнена полностью, но имеются не более одной негрубой ошибки и одного недочета, не более трех недочетов. Допускаются незначительные неточности при решении задач, решено 70% задач;

«Удовлетворительно» получают обучающиеся, у которых недостаточно высок уровень владения материалом. В процессе ответа на экзамене допускаются ошибки и затруднения при изложении материала. Обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой. Обучающийся затрудняется с правильной оценкой предложенной задачи, дает неполный ответ, решено 55% задач;

«Неудовлетворительно» получают обучающиеся, которые допускают значительные ошибки. Обучающийся имеет лишь начальную степень ориентации в материале. В работе число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы. Обучающийся дает неверную оценку ситуации, решено менее 50% задач.

6. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Максимальная сумма (100 баллов), набираемая студентом по дисциплине включает две составляющие:

- первая составляющая – оценка регулярности, своевременности и качества выполнения студентом учебной работы по изучению дисциплины в течение периода изучения дисциплины (семестра, или нескольких семестров) (сумма – не более 70 баллов). Баллы, характеризующие успеваемость студента по дисциплине, набираются им в течение всего периода обучения за изучение отдельных тем и выполнение отдельных видов работ.

- вторая составляющая – оценка знаний студента по результатам промежуточной аттестации (не более 30 –баллов).

Критерием оценки уровня сформированности компетенций в рамках учебной дисциплины «Теория языков программирования и методы трансляции» является экзамен.

Общий балл текущего и рубежного контроля складывается из следующих составляющих (Приложение).

Целью промежуточных аттестаций по дисциплине является оценка качества освоения дисциплины обучающимися.

Таблица 9

Результаты освоения учебной дисциплины, подлежащие проверке

Результаты обучения (компетенции)	Освоенные показатели оценки результатов обучения	Вид оценочного материала
Способен применять естественнонаучные и инженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности ОПК-1	<p>ИД-1опк-1 Знать основы высшей математики, физики, основы вычислительной техники и программирования</p> <p>ИД-2опк-1 Уметь решать стандартные профессиональные задачи с применением естественнонаучных и инженерных знаний, методов математического анализа и моделирования.</p> <p>ИД-3опк-1 Владеть навыками теоретического и экспериментального исследования объектов профессиональной деятельности.</p>	Типовые оценочные материалы для устного опроса, типовые тестовые задания, лабораторные работы, задания для самостоятельной работы (раздел 5)

7. Учебно-методическое обеспечение дисциплины

7.1. Нормативно-правовая база

- ГОСТ «Единая система программной документации».
- ГОСТ Р ИСО/МЭК ТО 15271-2002 «Руководство по применению ГОСТ Р ИСО/МЭК 12270 (Процессы жизненного цикла программных средств)».
- ГОСТ Р ИСО/МЭК ТО 12182-2010 «Классификация программных средств».
- ISO/IEC 14764:2006 «Разработка программного обеспечения. Процессы жизненного цикла программного обеспечения. Сопровождение».
- ГОСТ Р ИСО/МЭК 12119-2000 «Информационная технология. Пакеты программ. Требования к качеству и тестирование»
- ГОСТ 28195-89 «Оценка качества программных средств. Общие положения».
- ISO/IEC 25000:2005 «Технология программного обеспечения. Требования и оценка качества программного продукта. Руководство».
- ISO/IEC 25001:2014 «Программирование. Требования к качеству программного продукта и его оценка. Планирование и менеджмент».
- ISO/IEC 25010:2011 «Проектирование систем и разработка программного обеспечения. Требования к качеству систем и программного обеспечения и их оценка (SQuaRE). Модели качества систем и программного обеспечения».
- ISO/IEC 25012:2008 «Программная инженерия – Требования к качеству и оценке программного обеспечения. Модель качества данных».

11. ISO/IEC 25020:2007 «Разработка программного обеспечения. Требования к качеству и оценка качества программного продукта. Измерительная эталонная модель и руководство».

7.2. Основная литература

1. А.В.Ахо, М.С. Лам, Р.Сети, Д.Ульман. Компиляторы: принципы, технологии и инструментарий. 2-е изд. –М.: ООО «ИД Вильямс», 2011. Библиотека КБГУ, 2 экз.
2. Ю. Г. Карпов Теория и технология программирования. Основы построения трансляторов. Учеб. Пособие. Библиотека КБГУ, 1 экз.
3. Пентус А.Е. Математическая теория формальных языков [Электронный ресурс] / А.Е. Пентус, М.Р. Пентус. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 218 с. — 5-9556-0062-0. — Режим доступа: <http://www.iprbookshop.ru/52201.html>
4. Малявко А.А. Формальные языки и компиляторы [Электронный ресурс] : учебник / А.А. Малявко. — Электрон. текстовые данные. — Новосибирск: Новосибирский государственный технический университет, 2014. — 431 с. — 978-5-7782-2318-9. — Режим доступа: <http://www.iprbookshop.ru/47725.html>
5. Шульга Т.Э. Теория автоматов и формальных языков [Электронный ресурс] : учебное пособие / Т.Э. Шульга. — Электрон. текстовые данные. — Саратов: Саратовский государственный технический университет имени Ю.А. Гагарина, ЭБС АСВ, 2015. — 104 с. — 987-5-7433-2968-7. — Режим доступа: <http://www.iprbookshop.ru/76519.html>

7.3. Дополнительная литература

1. Дехтярь М. И. Введение в схемы, автоматы и алгоритмы. Национальный Открытый Университет «ИНТУИТ», 2016 г. 169 с.
2. Кауфман В.Ш. Языки программирования. Концепции и принципы [Электронный ресурс] / В.Ш. Кауфман. — Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 464 с. — 978-5-4488-0137-2. — Режим доступа: <http://www.iprbookshop.ru/64055.html>
3. Вирт Н. Построение компиляторов. ДМК Пресс, 2010.
4. Хопкрофт Дж., Мотвани Р., Ульман Дж. Введение в теорию автоматов, языков и вычислений. 2-е издание. - М.: Издательский дом "Вильямс", 2002. - 528 с.
5. Хантер Р. Основные концепции компиляторов. - М.: Издательский дом "Вильямс", 1984. - 256 с.
6. Теория языков программирования и методы трансляции: Методические указания к выполнению практических занятий. – СПб.: СЗТУ, 2003 – 36 с.

7.4. Периодические издания (газета, вестник, бюллетень, журнал)

1. Журнал «Компьютеры & Программы».
2. Журнал «Программирование».

7.5. Интернет-ресурсы

1. <http://iprbookshop.ru>
2. <https://www.codecademy.com/>
3. <https://itvdn.com/ru>
4. <http://lendwings.com/>

7.6. Современные профессиональные базы данных

1. База данных Science Index (РИНЦ) <http://elibrary.ru>
2. Национальная электронная библиотека РГБ <https://нэб.рф>
3. Крупнейшая единая база данных, содержащая аннотации и информацию о цитируемости рецензируемой научной литературы, со встроенными инструментами отслеживания, анализа и визуализации данных. www.scopus.com

4. Самая полная математическая база данных, охватывающая материалы с конца 19 века. zbMath содержит документы, журналы и книги по математике, статистике, информатике, а также машиностроению, физике, естественным наукам и др. www.zbmath.org (доступ открытый).

7.7. Методические указания по проведению различных учебных занятий и самостоятельной работы

Методические указания к лабораторным занятиям

Самостоятельная подготовка студентов к работе. Перед началом работы студенты должны четко представлять себе цель работы, знать схему, метод измерения, физическую сущность ожидаемых результатов. Должен быть подготовлен отчет, содержащий о порядке выполнения лабораторной работы. Студенты, не подготовившиеся к работе в соответствии с этими требованиями, к выполнению работы не допускаются.

Проведение эксперимента. Этот этап осуществляется в соответствии с методическими указаниями, которые содержатся в описании к каждой работе. Приступать к работам на стенде студент может начать только после ознакомления с теоретической частью и описания хода выполнения работы. Любые изменения в схеме проводятся при тщательной проверке схемы, для исключения короткого замыкания. Результаты выполнения проверяются преподавателем.

Составление отчета о проделанной работе. Отчёт должен содержать исчерпывающие данные, как о цели работы, так и о результатах в следующей последовательности: задание; схема установки и описание хода выполнения; результаты выполнения работы, включая рисунки, схемы, таблицы; общие выводы и заключение.

Текст отчета должен быть написан аккуратно и разборчиво от руки или представлен в виде распечатки, после компьютерной верстки. В обоих случаях текст должен представлять собой логическое изложение существа вопроса. Недопустимо приведение формул, таблиц без разъяснений всех обозначений и сокращений. Полученные зависимости должны сопровождаться теоретическим обоснованным объяснением причин, влияющих на их ход, для чего в процессе составления отчета студент обязан по литературным источникам ознакомиться с материалом, который был объектом его исследования в лаборатории. Без такого ознакомления с испытуемым методом студент не будет в состоянии дать правильный анализ процессов, происходящих в материале при эксперименте.

Защита лабораторной работы с представлением отчета. При сдаче отчета студенты должны показать понимание сущности проведенных исследований, объяснить полученные результаты и сделать выводы. При работе в лаборатории необходимо строго выполнять все правила техники безопасности и указания преподавателя.

Методические рекомендации по работе с литературой

Всю литературу можно разделить на учебники и учебные пособия, оригинальные научные монографические источники, научные публикации в периодической печати. Из них можно выделить литературу основную (рекомендуемую), дополнительную и литературу для углубленного изучения дисциплины.

Изучение дисциплины следует начинать с учебника, поскольку учебник – это книга, в которой изложены основы научных знаний по определенному предмету в соответствии с целями и задачами обучения, установленными программой.

При работе с литературой необходимо учитывать, что имеются различные виды чтения, и каждый из них используется на определенных этапах освоения материала.

Предварительное чтение направлено на выявление в тексте незнакомых терминов и поиск их значения в справочной литературе. В частности, при чтении указанной литературы необходимо подробнейшим образом анализировать понятия.

Сквозное чтение предполагает прочтение материала от начала до конца. Сквозное чтение литературы из приведенного списка дает возможность студенту сформировать свод основных понятий из изучаемой области и свободно владеть ими.

Выборочное – наоборот, имеет целью поиск и отбор материала. В рамках данного курса выборочное чтение, как способ освоения содержания курса, должно использоваться при подготовке к практическим занятиям по соответствующим разделам.

Аналитическое чтение – это критический разбор текста с последующим его конспектированием. Освоение указанных понятий будет наиболее эффективным в том случае, если при чтении текстов студент будет задавать к этим текстам вопросы. Часть из этих вопросов сформулирована в ФОС в перечне вопросов для собеседования. Перечень этих вопросов ограничен, поэтому важно не только содержание вопросов, но сам принцип освоения литературы с помощью вопросов к текстам.

Целью изучающего чтения является глубокое и всестороннее понимание учебной информации. Есть несколько приемов изучающего чтения:

1. Чтение по алгоритму предполагает разбиение информации на блоки: название; автор; источник; основная идея текста; фактический материал; анализ текста путем сопоставления имеющихся точек зрения по рассматриваемым вопросам; новизна.
2. Прием постановки вопросов к тексту имеет следующий алгоритм:
 - медленно прочитать текст, стараясь понять смысл изложенного;
 - выделить ключевые слова в тексте;
 - постараться понять основные идеи, подтекст и общий замысел автора.
3. Прием тезирования заключается в формулировании тезисов в виде положений, утверждений, выводов.

К этому можно добавить и иные приемы: прием реферирования, прием комментирования.

Важной составляющей любого солидного научного издания является список литературы, на которую ссылается автор. При возникновении интереса к какой-то обсуждаемой в тексте проблеме всегда есть возможность обратиться к списку относящейся к ней литературы. В этом случае вся проблема как бы разбивается на составляющие части, каждая из которых может изучаться отдельно от других. При этом важно не терять из вида общий контекст и не погружаться чрезмерно в детали, потому что таким образом можно не увидеть главного.

Подготовка к экзамену должна проводиться на основе лекционного материала, материала практических занятий с обязательным обращением к основным учебникам по курсу. Это позволит исключить ошибки в понимании материала, облегчит его осмысление, прокомментирует материал многочисленными примерами.

Методические рекомендации по организации самостоятельной работы

Самостоятельная работа обучающихся – способ активного, целенаправленного приобретения обучающимися новых для него знаний и умений без непосредственного участия в этом процесса преподавателей. Повышение роли самостоятельной работы обучающихся при проведении различных видов учебных занятий предполагает:

- оптимизацию методов обучения, внедрение в учебный процесс новых технологий обучения, повышающих производительность труда преподавателя, активное

использование информационных технологий, позволяющих обучающемуся в удобное для него время осваивать учебный материал;

- широкое внедрение компьютеризированного тестирования;
- совершенствование методики проведения практик и научно-исследовательской работы обучающихся, поскольку именно эти виды учебной работы в первую очередь готовят обучающихся к самостоятельному выполнению профессиональных задач;
- модернизацию системы курсового и дипломного проектирования, которая должна повышать роль обучающихся в подборе материала, поиске путей решения задач.

Самостоятельная работа приводит обучающихся к получению нового знания, упорядочению и углублению имеющихся знаний, формированию у него профессиональных навыков и умений. Самостоятельная работа выполняет ряд функций:

- развивающую;
- информационно-обучающую;
- ориентирующую и стимулирующую;
- воспитывающую;
- исследовательскую.

В рамках курса выполняются следующие виды самостоятельной работы:

- Проработка учебного материала (по конспектам, учебной и научной литературе);
- Выполнение разноуровневых заданий;
- Работа с тестами и вопросами для самопроверки;
- Выполнение итоговой контрольной работы.

Обучающимся рекомендуется с самого начала освоения курса работать с литературой и предлагаемыми заданиями в форме подготовки к очередному аудиторному занятию. При этом актуализируются имеющиеся знания, а также создается база для усвоения нового материала, возникают вопросы, ответы на которые обучающийся получает в аудитории.

Необходимо отметить, что некоторые задания для самостоятельной работы по курсу имеют определенную специфику. При освоении курса обучающийся может пользоваться библиотекой вуза, которая в полной мере обеспечена соответствующей литературой. Значительную помощь в подготовке к очередному занятию может оказать имеющийся в учебно-методическом комплексе краткий конспект лекций. Он же может использоваться и для закрепления полученного в аудитории материала. Самостоятельная работа обучающихся предусмотрена учебным планом и выполняется в обязательном порядке. Задания предложены по каждой изучаемой теме и могут готовиться индивидуально или в группе. По необходимости обучающийся может обращаться за консультацией к преподавателю. Выполнение заданий контролируется и оценивается преподавателем.

Для успешной организации самостоятельной работы все активнее применяются разнообразные образовательные ресурсы в сети Интернет: системы тестирования по различным областям, виртуальные лекции, лаборатории, при этом пользователю достаточно иметь компьютер и подключение к Интернету для того, чтобы связаться с преподавателем, решать вычислительные задачи и получать знания. Использование сетей усиливает роль самостоятельной работы обучающихся и позволяет кардинальным образом изменить методику преподавания.

Обучающийся может получать все задания и методические указания через сервер, что дает ему возможность привести в соответствие личные возможности с необходимыми для выполнения работ трудозатратами. Обучающийся имеет возможность выполнять работу дома или в аудитории. Большое воспитательное и образовательное значение в самостоятельном учебном труде обучающийся имеет самоконтроль. Самоконтроль возбуждает и поддерживает внимание и интерес, повышает активность памяти и мышления, позволяет обучающемуся своевременно обнаружить и устранить допущенные ошибки и недостатки, объективно

определить уровень своих знаний, практических умений. Самое доступное и простое средство самоконтроля с применением информационно-коммуникационных технологий - это ряд тестов «on-line», которые позволяют в режиме реального времени определить свой уровень владения предметным материалом, выявить свои ошибки и получить рекомендации по самосовершенствованию.

Методические рекомендации для подготовки к экзамену

Экзамен является формой итогового контроля знаний и умений обучающихся по данной дисциплине, полученных на лекциях, лабораторных занятиях и в процессе самостоятельной работы. Основой для определения оценки служит уровень усвоения обучающимися материала, предусмотренного данной рабочей программой. К экзамену допускаются студенты, набравшие 36 и более баллов по итогам текущего и промежуточного контроля. На экзамене студент может набрать от 15 до 30 баллов.

В период подготовки к экзамену обучающиеся вновь обращаются к учебно-методическому материалу и закрепляют промежуточные знания.

Подготовка обучающегося к экзамену включает три этапа:

- самостоятельная работа в течение семестра;
- непосредственная подготовка в дни, предшествующие экзамену по темам курса;
- подготовка к ответу на экзаменационные вопросы.

При подготовке к экзамену обучающимся целесообразно использовать материалы лекций, учебно-методические комплексы, нормативные документы, основную и дополнительную литературу.

На экзамен выносится материал в объеме, предусмотренном рабочей программой учебной дисциплины за семестр. Экзамен проводится в письменной / устной форме.

При проведении экзамена в письменной (устной) форме, ведущий преподаватель составляет экзаменационные билеты, которые включают в себя: тестовые задания; теоретические задания; задачи или ситуации. Формулировка теоретических задания совпадает с формулировкой перечня экзаменационных вопросов, доведенных до сведения обучающихся накануне экзаменационной сессии. Содержание вопросов одного билета относится к различным разделам программы с тем, чтобы более полно охватить материал учебной дисциплины.

В аудитории, где проводится устный экзамен, должно одновременно находиться не более шести студентов на одного преподавателя, принимающего экзамен. На подготовку ответа на билет на экзамене отводится 40 минут.

При проведении письменного экзамена на работу отводится 60 минут.

Результат устного (письменного) экзамена выражается оценками:

Оценка «отлично» – от 91 до 100 баллов – теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы. Все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному. На экзамене студент демонстрирует глубокие знания предусмотренного программой материала, умеет четко, лаконично и логически последовательно отвечать на поставленные вопросы.

Оценка «хорошо» – от 81 до 90 баллов – теоретическое содержание курса освоено, необходимые практические навыки работы сформированы, выполненные учебные задания содержат незначительные ошибки. На экзамене студент демонстрирует твердые знания основного (программного) материала,

умеет четко, грамотно, без существенных неточностей отвечать на поставленные вопросы.

Оценка «удовлетворительно» – от 61 до 80 баллов – теоретическое содержание курса освоено не полностью, необходимые практические навыки работы сформированы частично, выполненные учебные задания содержат грубые ошибки. На экзамене студент демонстрирует знание только основного материала, ответы содержат неточности, слабо аргументированы, нарушена последовательность изложения материала.

Оценка «неудовлетворительно» – от 36 до 60 баллов – теоретическое содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к существенному повышению качества выполнения учебных заданий. На экзамене студент демонстрирует незнание значительной части программного материала, существенные ошибки в ответах на вопросы, неумение ориентироваться в материале, незнание основных понятий дисциплины.

8. Материально-техническое обеспечение дисциплины

8.1. Требования к материально-техническому обеспечению

Минимально необходимый для реализации ОПОП перечень материально-технического обеспечения включает в себя: лекционные аудитории (оборудованные видеопроекционным оборудованием для презентаций, средствами звуковоспроизведения, экраном и имеющие выход в сеть Интернет), помещения для проведения семинарских и практических занятий (оборудованные учебной мебелью), компьютерные классы и др. (в соответствии с ФГОС и учебным планом).

Во время самостоятельной работы студенты используют компьютерные классы института информатики, электроники и компьютерных технологий, электронные читальные залы КБГУ и домашние компьютеры.

Лицензионное программное обеспечение, используемое для проведения лекционных и лабораторных занятий

1. Microsoft Windows 10.
2. Microsoft Office 2016.
3. Visual Studio 2019.

8.2. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Для студентов с ограниченными возможностями здоровья созданы специальные условия для получения образования. В целях доступности получения высшего образования по образовательным программам инвалидами и лицами с ограниченными возможностями здоровья университетом обеспечивается (аудитория для самостоятельной работы и коллективного пользования специальными техническими средствами для обучения инвалидов и лиц с ОВЗ в КБГУ, аудитория № 145 Главный корпус КБГУ):

1. Альтернативная версия официального сайта в сети «Интернет» для слабовидящих;
2. Для инвалидов с нарушениями зрения (слабовидящие, слепые):
 - присутствие ассистента, оказывающего обучающемуся необходимую помощь, дублирование вслух справочной информации о расписании учебных занятий; наличие средств для усиления остаточного зрения, брайлевской компьютерной техники, видеоувеличителей, программ невизуального доступа к информации, программ-синтезаторов речи и других технических средств приема-передачи учебной информации в доступных формах для студентов с нарушениями зрения;
 - задания для выполнения на экзамене зачитываются ассистентом;
 - письменные задания выполняются на бумаге, надиктовываются ассистенту

обучающимся;

3. Для инвалидов и лиц с ограниченными возможностями здоровья по слуху (слабослышащие, глухие):

- на зачете/экзамене присутствует ассистент, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (он помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе записывая под диктовку);
- зачет/экзамен проводится в письменной форме;

4. Для инвалидов и лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, созданы материально-технические условия, обеспечивающие возможность беспрепятственного доступа обучающихся в учебные помещения, объекты питания, туалетные и другие помещения университета, а также пребывания в указанных помещениях (наличие расширенных дверных проемов, поручней и других приспособлений).

- письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;
- по желанию студента экзамен проводится в устной форме.

Обучающиеся из числа лиц с ограниченными возможностями здоровья обеспечены электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

9. Лист переутверждения рабочей программы дисциплины

Рабочая программа:

одобрена на 2023/2024 учебный год. Протокол № _____ заседания кафедры от
«__» _____ 2023 г.

В рабочую программу внесены следующие изменения:

Разработчик программы _____

Зав. кафедрой _____

Приложение

Распределение баллов текущего и рубежного контроля

№п/п	Вид контроля	Сумма баллов			
		Общая сумма	1-я точка	2-я точка	3-я точка
1	Посещение занятий	до 10 б.	до 3 б.	до 3б.	до 4б.
2	Текущий контроль:	до 28 б.	до 9 б.	до 9 б.	до 10 б.
	Выполнение практических работ	до 18 б.	до 6 б.	до 6 б.	до 6 б.
	Выполнение самостоятельных заданий	от 0 до 10 б.	от 0 до 3 б.	от 0 до 3 б.	от 0 до 4 б.
3	Рубежный контроль	до 27 баллов	до 9 б.	до 9 б.	до 9 б.
	тестирование	от 0 до 15 б.	от 0 до 5 б.	от 0 до 5 б.	от 0 до 5 б.
	коллоквиум	от 0 до 12 б.	от 0 до 4 б.	от 0 до 4 б.	от 0 до 4 б.
4	Итого сумма текущего и рубежного контроля	до 70 б.	до 23 б.	до 23 б.	до 24 б.