ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Иркутский государственный университет путей сообщения»
(ФГБОУ ВО ИрГУПС)

УТВЕРЖДЕНА
приказом и.о. ректора
от «08» мая 2020 г. № 267-1

Б1.О.31 Теория языков программирования и методы трансляции

рабочая программа дисциплины

Специальность/направление подготовки – 09.03.04 Программная инженерия
Специализация/профиль – Разработка программно-информационных систем
Квалификация выпускника – Бакалавр
Форма и срок обучения – очная форма 4 года
Кафедра-разработчик программы – Информационные системы и защита информации

Общая трудоемкость в з.е. – 5
Часов по учебному плану (УП) – 180

Формы промежуточной аттестации
очная форма обучения:
экзамен 7 семестр, курсовая работа 7 семестр

Очная форма обучения **Распределение часов дисциплины по семестрам**

Семестр	7	**Итого**
Вид занятий	Часов по УП	**Часов по УП**
Аудиторная контактная работа по видам учебных занятий/ в т.ч. в форме ПП*	56	**56**
– лекции	28	**28**
– практические (семинарские)		
– лабораторные	28	**28**
Самостоятельная работа	88	**88**
Экзамен	36	**36**
Итого	180	**180**

ИРКУТСК

Электронный документ выгружен из ЕИС ФГБОУ ВО ИрГУПС и соответствует оригиналу
Подписант ФГБОУ ВО ИрГУПС Трофимов Ю.А.
00920FD815CE68F8C4CA795540563D259C с 07.02.2024 05:46 по 02.05.2025 05:46 GMT+03:00
Подпись соответствует файлу документа

Рабочая программа дисциплины разработана в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по направлению подготовки 09.03.04 Программная инженерия, утвержденным Приказом Министерства науки и высшего образования Российской Федерации от 19.09.2017 № 920.

Программу составил(и):
ассистент, В.М.Замятин
к.т.н., доцент, Ю.Н. Шишкин

Рабочая программа рассмотрена и одобрена для использования в учебном процессе на заседании кафедры «Информационные системы и защита информации», протокол от «29» апреля 2020 г. № 11

Зав. кафедрой, д.т.н., доцент Л.В. Аршинский

1 ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ	
1.1 Цели дисциплины	
1	формирование у обучающихся знаний по теории формальных языков и способам их моделирования
2	формирование у обучающихся теоретических знаний о методах трансляции и компиляции, способах и технологиях их применения
1.2 Задачи дисциплины	
1	изучение основ теории формальных языков и методов трансляции
2	рассмотрение проблематики методов трансляции и современных технологий разработки трансляторов
1.3 Цель воспитания и задачи воспитательной работы в рамках дисциплины	
Профессионально-трудовое воспитание обучающихся	
Цель профессионально-трудового воспитания – формирование у обучающихся осознанной профессиональной ориентации, понимания общественного смысла труда и значимости его для себя лично, ответственного, сознательного и творческого отношения к будущей деятельности, профессиональной этики, способности предвидеть изменения, которые могут возникнуть в профессиональной деятельности, и умению работать в изменённых, вновь созданных условиях труда. Цель достигается по мере решения в единстве следующих задач: – формирование сознательного отношения к выбранной профессии; – воспитание чести, гордости, любви к профессии, сознательного отношения к профессиональному долгу, понимаемому как личная ответственность и обязанность; – формирование психологии профессионала; – формирование профессиональной культуры, этики профессионального общения; – формирование социальной компетентности и другие задачи, связанные с имиджем профессии и авторитетом транспортной отрасли	

2 МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП	
Блок/часть ОПОП	Блок 1. Дисциплины / Обязательная часть
2.1 Дисциплины и практики, на которых основывается изучение данной дисциплины	
1	Б1.О.13 Математическая логика и теория алгоритмов
2	Б1.О.18 Программирование
3	Б1.О.21 Операционные системы
4	Б1.О.23 Архитектура ЭВМ
5	Б1.О.26 Объектно-ориентированное программирование
6	Б1.О.29 Вычислительные алгоритмы
7	Б1.О.32 Машинно-зависимые языки программирования
8	Б1.О.33 Схемотехнические основы программно-вычислительных систем
9	Б1.О.34 Теория информации
10	Б2.О.03(П) Производственная - технологическая (проектно-технологическая) практика
2.2 Дисциплины и практики, для которых изучение данной дисциплины необходимо как предшествующее	
1	Б1.О.25 Тестирование и отладка программного обеспечения
2	Б1.О.27 Надежность программного обеспечения
3	Б2.О.04(Пд) Производственная - преддипломная практика
4	Б3.01(Д) Выполнение выпускной квалификационной работы
5	Б3.02(Д) Защита выпускной квалификационной работы

3 ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ТРЕБОВАНИЯМИ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ		
Код и наименование компетенции	**Код и наименование индикатора достижения компетенции**	**Планируемые результаты обучения**
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования	ОПК-6.1 Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий	Знать: основные языки программирования и работы с базами данных, их логику построения и принципы функционирования
		Уметь: применять языки программирования для работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов
		Владеть: навыками разработки алгоритмов и программ, пригодных для практического применения в профессиональной области

к проектированию, конструированию и тестированию программных продуктов	ОПК-6.2 Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ	Знать: технологии применения трансляторов и компиляторов при решении прикладных задач
		Уметь: применять технологии конструирования трансляторов и компиляторов для решения прикладных задач
		Владеть: навыками разработки трансляторов и конструирования компиляторов
	ОПК-6.3 Имеет навыки программирования, отладки и тестирования прототипов программно-технических комплексов задач	Знать: методы разработки алгоритмов и программного обеспечения
		Уметь: разрабатывать оригинальные алгоритмы и программные продукты на нескольких языках программирования
		Владеть: навыками анализа, программирования, отладки и тестирования прототипов программно-технических комплексов для решения профессиональных задач
ОПК-7 Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	ОПК-7.1 Понимает основные концепции, принципы, теории и факты, связанные с информатикой, операционными системами, построением вычислительных систем	Знать: классификацию языков программирования, средств исполнения программ, принципы работы трансляторов
		Уметь: применять на практике технологии программирования, навыки программирования при создании разнообразных программ
		Владеть: навыками программирования и методами синтаксического анализа современных языков программирования
	ОПК-7.2 Применяет на практике основные концепции, теории информационных процессов передачи, хранения и преобразования сообщений в технических системах, теории сигналов, теории информации и кодирования	Знать: основные языки программирования и работы с базами данных, операционные системы и оболочки
		Уметь: обосновывать выбор технологий конструирования компиляторов
		Владеть: навыками анализа технологий конструирования компиляторов; применять языки программирование для разработки программного обеспечения для профессиональных задач
	ОПК-7.3 Демонстрирует навыки решения задач с использованием основ информатики, концепции теории информации для решения задач профессиональной деятельности	Знать: основные платформы, технологии и инструментальные программно-аппаратные средства
		Уметь: реализовывать программные средства с использованием технологии конструирования компиляторов
		Владеть: навыками разработки программного обеспечения, включая особенности проектирования, отладки, проверки работоспособности и модификации ПО

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код	Наименование разделов, тем и видов работ	Очная форма					*Код индикатора достижения компетенции
		Семестр	Часы				
			Лек	Пр	Лаб	СР	
1.0	**Раздел 1. Понятие трансляции, компиляции и интерпретации. Языки высокого уровня, байт-код и ассемблеры.**						
1.1	Проработка лекционного материала и подготовка к практическим занятиям раздела 1	7				4	ОПК-6.1 ОПК-6.2 ОПК-7.1
1.2	Трансляция, компиляция и интерпретация – основные понятия. Формальные языки и грамматики.	7	2				ОПК-6.1 ОПК-6.2 ОПК-7.1
1.3	Основные этапы компиляции. Языки высокого уровня, байт-код и ассемблеры	7	2				ОПК-6.1 ОПК-6.2

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ							
Код	**Наименование разделов, тем и видов работ**	**Очная форма**					***Код индикатора достижения компетенции**
		Семестр	**Часы**				
			Лек	**Пр**	**Лаб**	**СР**	
							ОПК-7.1
1.4	Проектирование учебного компилятора	7			4		ОПК-6.1 ОПК-6.2 ОПК-7.1
1.5	Проектирование учебного компилятора для курсовой работы	7				4	ОПК-6.1 ОПК-6.2 ОПК-7.1
2.0	**Раздел 2. Методы, модели и алгоритмы лексического анализа.**						
2.1	Проработка лекционного материала и подготовка к практическим занятиям раздела 2	7				6	ОПК-6.1 ОПК-6.2 ОПК-6.3
2.2	Задача лексического анализа. Регулярные языки, регулярные грамматики.	7	2				ОПК-6.1 ОПК-6.2 ОПК-6.3
2.3	Конечные автоматы и регулярные выражения.	7	2				ОПК-6.1 ОПК-6.2 ОПК-6.3
2.4	Проектирование и разработка лексического анализатора учебного компилятора	7			4		ОПК-6.1 ОПК-6.2 ОПК-6.3
2.5	Проектирование и разработка лексического анализатора для курсовой работы	7				6	ОПК-6.1 ОПК-6.2 ОПК-6.3
3.0	**Раздел 3. Методы, модели и алгоритмы синтаксического анализа.**						
3.1	Проработка лекционного материала и подготовка к практическим занятиям раздела 3	7				8	ОПК-7.1 ОПК-7.2 ОПК-7.3
3.2	Задача синтаксического анализа. КС-грамматики.	7	2				ОПК-7.1 ОПК-7.2 ОПК-7.3
3.3	Синтаксические распознаватели с возвратом.	7	2				ОПК-7.1 ОПК-7.2 ОПК-7.3
3.4	Нисходящие синтаксические распознаватели КС-языков без возврата	7	2				ОПК-7.1 ОПК-7.2 ОПК-7.3
3.5	Восходящие синтаксические распознаватели КС-языков без возврата	7	2				ОПК-7.1 ОПК-7.2 ОПК-7.3
3.6	Синтаксические распознаватели на основе грамматик предшествования	7	2				ОПК-7.1 ОПК-7.2 ОПК-7.3
3.7	Проектирование и разработка синтаксического анализатора на основе LL(1)-грамматики	7			6		ОПК-7.1 ОПК-7.2 ОПК-7.3
3.8	Проектирование и разработка синтаксического анализатора на основе грамматики операторного предшествования	7				8	ОПК-7.1 ОПК-7.2 ОПК-7.3
4.0	**Раздел 4. Методы, модели и алгоритмы семантического анализа. Синтаксически-управляемая трансляция.**						
4.1	Проработка лекционного материала и подготовка к практическим занятиям раздела 4	7				8	ОПК-6.1 ОПК-7.2 ОПК-7.3
4.2	Методы, модели и алгоритмы семантического анализа. Граф потока управления и граф потоков данных.	7	2				ОПК-6.1 ОПК-7.2 ОПК-7.3

4 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Код	Наименование разделов, тем и видов работ	Очная форма					*Код индикатора достижения компетенции
		Семестр	Часы				
			Лек	Пр	Лаб	СР	
4.3	Синтаксически-управляемая трансляция – понятия, возможности и алгоритмы применения	7	2				ОПК-6.1 ОПК-7.2 ОПК-7.3
4.4	Проектирование и разработка учебного семантического анализатора	7			4		ОПК-6.1 ОПК-7.2 ОПК-7.3
5.0	**Раздел 5. Методы и модели создания внутреннего представления программы. Современные подходы и критерии оптимизации программ при компиляции.**						
5.1	Проработка лекционного материала и подготовка к практическим занятиям раздела 5	7				6	ОПК-7.1 ОПК-7.2 ОПК-7.3
5.2	Внутреннее представление программы – основные понятия, виды, алгоритмы генерации.	7	2				ОПК-7.1 ОПК-7.2 ОПК-7.3
5.3	Современные подходы и критерии оптимизации программ при компиляции.	7	2				ОПК-7.1 ОПК-7.2 ОПК-7.3
5.4	Разработка алгоритма генерации и оптимизации триад для учебного компилятора	7			4		ОПК-7.1 ОПК-7.2 ОПК-7.3
5.5	Проектирование и разработка подсистемы оптимизации триад для курсовой работы	7				6	ОПК-7.1 ОПК-7.2 ОПК-7.3
6.0	**Раздел 6. Методы и алгоритмы генерации объектного кода.**						
6.1	Проработка лекционного материала и подготовка к практическим занятиям раздела 6	7				8	ОПК-6.1 ОПК-7.2 ОПК-7.3
6.2	Задача генерации объектного кода и способы ее решения.	7	2				ОПК-6.1 ОПК-7.2 ОПК-7.3
6.3	Проектирование и реализация модуля оптимизации внутреннего представления программы и генерации объектного кода	7			6		ОПК-6.1 ОПК-7.2 ОПК-7.3
6.4	Проектирование и разработка подсистемы кодогенерации для курсовой работы	7				8	ОПК-6.1 ОПК-7.2 ОПК-7.3
	Форма промежуточной аттестации – экзамен	7	36				ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3
	Курсовая работа	7				16	ОПК-6.1 ОПК-7.2 ОПК-7.3
	Итого часов (без учёта часов на промежуточную аттестацию)		28		28	88	

5 ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине оформлен в виде приложения № 1 к рабочей программе дисциплины и размещен в электронной информационно-образовательной среде Университета, доступной обучающемуся через его личный кабинет

6 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1 Учебная литература **6.1.1 Основная литература**		
	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.1.1	Малявко, А. А. Формальные языки и компиляторы : учебное пособие / А. А. Малявко. Новосибирск : Новосибирский государственный технический университет, 2014. - 431с. - Текст: электронный. - URL: https://biblioclub.ru/index.php?page=book&id=436055 (дата обращения: 23.05.2024)	Онлайн
6.1.2 Дополнительная литература		
	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.2.1	Малявко, А. А. Системное программное обеспечение: формальные языки и методы трансляции : учебное пособие / А. А. Малявко. Новосибирск : Новосибирский государственный технический университет, 2010. - 104с. - Текст: электронный. - URL: https://biblioclub.ru/index.php?page=book&id=228974 (дата обращения: 23.05.2024)	Онлайн
6.1.2.2	Малявко, А. А. Системное программное обеспечение: формальные языки и методы трансляции : учебное пособие / А. А. Малявко. Новосибирск : Новосибирский государственный технический университет, 2011. - 160с. - Текст: электронный. - URL: https://biblioclub.ru/index.php?page=book&id=228973 (дата обращения: 23.05.2024)	Онлайн
6.1.2.3	Малявко, А. А. Системное программное обеспечение: формальные языки и методы трансляции : учебное пособие / А. А. Малявко. Новосибирск : Новосибирский государственный технический университет, 2012. - 120с. - Текст: электронный. - URL: https://biblioclub.ru/index.php?page=book&id=228888 (дата обращения: 23.05.2024)	Онлайн
6.1.3 Учебно-методические разработки (в т. ч. для самостоятельной работы обучающихся)		
	Библиографическое описание	Кол-во экз. в библиотеке/ онлайн
6.1.3.1	Замятин, В.М. Методические указания по изучению дисциплины Б1.О.31 Теория языков программирования и методы трансляции по направлению подготовки 09.03.04 Программная инженерия, профиль Разработка программно-информационных систем (по отрасли или в сфере профессиональной деятельности) / В.М. Замятин ; ИрГУПС. – Иркутск : ИрГУПС, 2023. – 14 с. - Текст: электронный. - URL: https://www.irgups.ru/eis/for_site/umkd_files/mu_2626_1398_2020_1_signed.pdf	Онлайн
6.2 Ресурсы информационно-телекоммуникационной сети «Интернет»		
6.2.1	Электронно-библиотечная система «Университетская библиотека онлайн», https://biblioclub.ru/	
6.3 Программное обеспечение и информационные справочные системы **6.3.1 Базовое программное обеспечение**		
6.3.1.1	Microsoft Windows Professional 10, государственный контракт от 20.07.2021 № 0334100010021000013-01	
6.3.1.2	Microsoft Office Russian 2010, государственный контракт от 20.07.2021 № 0334100010021000013-01	
6.3.1.3	FoxitReader, свободно распространяемое программное обеспечение http://free-software.com.ua/pdf-viewer/foxit-reader/	
6.3.1.4	Adobe Acrobat Reader DC свободно распространяемое программное обеспечение https://get.adobe.com/ru/reader/enterprise/	
6.3.1.5	Яндекс. Браузер. Прикладное программное обеспечение общего назначения, Офисные приложения, лицензия – свободно распространяемое программное обеспечение по лицензии BSD License	
6.3.2 Специализированное программное обеспечение		
6.3.2.1	Visual Studio Code, образовательная лицензия, https://code.visualstudio.com/license NetBeans IDE, свободная лицензия Apache License 2.0 https://www.apache.org/licenses/ Python 3.9, свободно распространяемое программное обеспечение https://docs.python.org/3/license.html	
6.3.2.2	Dev-C++, свободная интегрированная среда разработки приложений для языков программирования C/C++, https://code-live.ru/post/dev-cpp-free-cpp-ide-for-windows/	

6.3.2.4	Dev-C , свободная интегрированная среда разработки приложений для языков программирования C/C , https://code-live.ru/post/dev-cpp-free-cpp-ide-for-windows/
6.3.3 Информационные справочные системы	
6.3.3.1	Не предусмотрены
6.4 Правовые и нормативные документы	
6.4.1	Не предусмотрены

7 ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ УЧЕБНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

1	Корпуса А, Б, В, Г, Д, Е ИрГУПС находятся по адресу г. Иркутск, ул. Чернышевского, д. 15; корпус Л ИрГУПС находится – по адресу г. Иркутск, ул. Лермонтова, д.80
2	Учебная аудитория Д-417*для проведения лекционных и практических занятий, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор, экран, (ноутбук переносной). Для проведения занятий имеются учебно-наглядные пособия (презентации).
3	Учебная аудитория Д-518* для проведения лекционных и практических занятий, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор, экран, (ноутбук переносной). Для проведения занятий имеются учебно-наглядные пособия (презентации).
4	Класс А-401 "Деловых игр" для проведения практических занятий, лабораторных работ, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор, экран, компьютеры с подключением к сети Интернет, обеспечивающие доступ в электронную информационно-образовательную среду ИрГУПС. Для проведения занятий имеются учебно-наглядные пособия (презентации).
5	Компьютерный класс А-513 для проведения практических занятий, лабораторных работ, групповых и индивидуальных консультаций, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации. Основное оборудование: специализированная мебель, мультимедиапроектор (переносной), экран (переносной), компьютеры с подключением к сети Интернет, обеспечивающие доступ в электронную информационно-образовательную среду ИрГУПС.
6	Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой, подключенной к информационно-телекоммуникационной сети «Интернет», и обеспечены доступом в электронную информационно-образовательную среду ИрГУПС. Помещения для самостоятельной работы обучающихся: – читальные залы; – учебные залы вычислительной техники А-401, А-509, А-513, А-516, Д-501, Д-503, Д-505, Д-507; – помещения для хранения и профилактического обслуживания учебного оборудования – А-521

8 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Вид учебной деятельности	Организация учебной деятельности обучающегося
Лекция	Лекция (от латинского «lection» – чтение) – вид аудиторных учебных занятий. Лекция: закладывает основы научных знаний в систематизированной, последовательной, обобщенной форме; раскрывает состояние и перспективы развития соответствующей области науки и техники; концентрирует внимание обучающихся на наиболее сложных, узловых вопросах; стимулирует познавательную активность обучающихся. Во время лекционных занятий обучающийся должен уметь сконцентрировать внимание на изучаемых проблемах и включить в работу все виды памяти: словесную, образную и моторно-двигательную. Для этого весь материал, излагаемый преподавателем, обучающемуся необходимо конспектировать. На полях конспекта следует помечать вопросы, выделенные обучающимся для консультации с преподавателем. Выводы, полученные в виде формул, рекомендуется в конспекте подчеркивать или обводить рамкой, чтобы лучше запоминались. Полезно составить краткий справочник, содержащий определения важнейших понятий лекции. К каждому занятию следует разобрать материал предыдущей лекции. Изучая материал по учебнику или конспекту лекций, следует переходить к следующему вопросу только в том случае, когда хорошо усвоен предыдущий вопрос. Ряд вопросов дисциплины может быть вынесен на самостоятельное изучение. Такое задание требует оперативного выполнения. В конспекте лекций необходимо оставить место для освещения упомянутых вопросов. Обозначить вопросы, термины, материал, который

	вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удается разобраться в материале, то необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии
Практическое занятие	Практическое занятие – вид аудиторных учебных занятий, целенаправленная форма организации учебного процесса, при реализации которой обучающиеся по заданию и под руководством преподавателя выполняют практические задания. Практические задания направлены на углубление научно-теоретических знаний и овладение определенными методами работы, в процессе которых вырабатываются умения и навыки выполнения тех или иных учебных действий в данной сфере науки. Практические занятия развивают научное мышление и речь, позволяют проверить знания обучающихся, выступают как средства оперативной обратной связи; цель практических занятий – углублять, расширять, детализировать знания, полученные на лекции, в обобщенной форме и содействовать выработке навыков профессиональной деятельности. На практических занятиях подробно рассматриваются основные вопросы дисциплины, разбираются основные типы задач. К каждому практическому занятию следует заранее самостоятельно выполнить домашнее задание и выучить лекционный материал к следующей теме. Систематическое выполнение домашних заданий обязательно и является важным фактором, способствующим успешному усвоению дисциплины
Лабораторная работа	Основной целью лабораторных работ является теоретическое обоснование, наглядное и/или экспериментальное подтверждение и/или проверка существенных теоретических положений (законов, закономерностей) анализ существующих методик и методов их реализации и т.д. Они занимают преимущественное место при изучении дисциплин обязательной части и части, формируемой участниками образовательных отношений Блока 1. Исходя из цели, содержанием лабораторных работ могут быть: - экспериментальная проверка формул, методик расчета; - проведение натурных измерений свойств, рабочих параметров, режимов работы при помощи лабораторного оборудования и/или стендов и макетов; - ознакомление, анализ и теоретические выкладки по устройству, принципу действия и способам обслуживания аппаратов, деталей машин, механизмов, процессов, протекающих в них при этом и т.д.; - наглядная графическая интерпретация чертежей, схем, объемных поверхностей и т.д., воспроизводимых с помощью специализированного программного обеспечения; - имитационное моделирование процессов, протекающих в сложных химических, физических, механических, электрических и пр. объектах; - наглядное представление о работе персонала конкретной организации или подразделения ОАО «РЖД» посредством моделирования штатных и внештатных ситуаций в виртуальных специализированных АРМ (автоматизированных рабочих мест); - установление и подтверждение закономерностей (путем сравнения проведенного эксперимента и рассчитанных значений) и т.д.; - ознакомление с методиками проведения экспериментов, наглядным устройством стенд-макетов и пр.; - установление свойств веществ, их качественных и количественных характеристик; - анализ различных характеристик процессов, в том числе производственных и иных процессов; - расчет параметров различных явлений и процессов, смоделировать которые не возможно в реальных условиях (например, чрезвычайные ситуации и пр.); - наблюдение развития явлений, процессов и др. Допускается иное содержание лабораторных работ, если это будет способствовать реализации целей и задач дисциплины и формированию соответствующих компетенций. По характеру выполняемых лабораторных работ возможны: - ознакомительные работы, используемые для закрепления изученного теоретического материалы; - аналитические работы, используемые для получения новой информации на основе формализованных методов; - творческие работы, ориентированные на самостоятельный выбор подходов решения задач. Прежде, чем приступить к лабораторным занятиям, обучающимся необходимо повторить теоретический материал по теме работы. Каждая лабораторная работа оснащена методическими указаниями, разработанными преподавателями, ведущими дисциплину
Самостоятельная работа	Обучение по дисциплине «Теория языков программирования и методы трансляции» предусматривает активную самостоятельную работу обучающегося. В разделе 4 рабочей программы, который называется «Структура и содержание дисциплины», все часы самостоятельной работы расписаны по темам и вопросам, а также указана необходимая учебная литература: обучающийся изучает учебный материал, разбирает примеры и решает

	разноуровневые задачи в рамках выполнения как общих домашних заданий, так и индивидуальных домашних заданий (ИДЗ) и других видов работ, предусмотренных рабочей программой дисциплины. При выполнении домашних заданий обучающемуся следует обратиться к задачам, решенным на предыдущих практических занятиях, решенным домашним работам, а также к примерам, приводимым лектором. Если этого будет недостаточно для выполнения всей работы можно дополнительно воспользоваться учебными пособиями, приведенными в разделе 6.1 «Учебная литература». Если, несмотря на изученный материал, задание выполнить не удается, то в обязательном порядке необходимо посетить консультацию преподавателя, ведущего практические занятия, и/или консультацию лектора. Домашние задания, индивидуальные домашние задания и другие работы, предусмотренные рабочей программой дисциплины должны быть выполнены обучающимся в установленные преподавателем сроки в соответствии с требованиями к оформлению текстовой и графической документации, сформулированным в Положении «Требования к оформлению текстовой и графической документации. Нормоконтроль»
Комплекс учебно-методических материалов по всем видам учебной деятельности, предусмотренным рабочей программой дисциплины (модуля), размещен в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет	

Приложение № 1 к рабочей программе

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

для проведения текущего контроля успеваемости и промежуточной аттестации

1. Общие положения

Фонд оценочных средств (ФОС) является составной частью нормативно-методического обеспечения системы оценки качества освоения обучающимися образовательной программы.

Фонд оценочных средств предназначен для использования обучающимися, преподавателями, администрацией ИрГУПС, а также сторонними образовательными организациями для оценивания качества освоения образовательной программы и уровня сформированности компетенций у обучающихся.

Задачами ФОС являются:

– оценка достижений обучающихся в процессе изучения дисциплины;

– обеспечение соответствия результатов обучения задачам будущей профессиональной деятельности через совершенствование традиционных и внедрение инновационных методов обучения в образовательный процесс;

– самоподготовка и самоконтроль обучающихся в процессе обучения.

Фонд оценочных средств сформирован на основе ключевых принципов оценивания: валидность, надежность, объективность, эффективность.

Для оценки уровня сформированности компетенций используется трехуровневая система:

– минимальный уровень освоения, обязательный для всех обучающихся по завершению освоения образовательной программы; дает общее представление о виде деятельности, основных закономерностях функционирования объектов профессиональной деятельности, методов и алгоритмов решения практических задач;

– базовый уровень освоения, превышение минимальных характеристик сформированности компетенций; позволяет решать типовые задачи, принимать профессиональные и управленческие решения по известным алгоритмам, правилам и методикам;

– высокий уровень освоения, максимально возможная выраженность характеристик компетенций; предполагает готовность решать практические задачи повышенной сложности, нетиповые задачи, принимать профессиональные и управленческие решения в условиях неполной определенности, при недостаточном документальном, нормативном и методическом обеспечении.

2. Перечень компетенций, в формировании которых участвует дисциплина.
Программа контрольно-оценочных мероприятий.
Показатели оценивания компетенций, критерии оценки

Дисциплина «Теория языков программирования и методы трансляции» участвует в формировании компетенций:

ОПК-6. Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов

ОПК-7. Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой

Программа контрольно-оценочных мероприятий очная форма обучения

№	Наименование контрольно-оценочного мероприятия	Объект контроля	Код индикатора достижения компетенции	Наименование оценочного средства (форма проведения*)
7 семестр				
1.0	**Раздел 1. Понятие трансляции, компиляции и интерпретации. Языки высокого уровня, байт-код и ассемблеры**			
1.1	Текущий контроль	Проработка лекционного материала и подготовка к практическим занятиям раздела 1	ОПК-6.1 ОПК-6.2 ОПК-7.1	Конспект (письменно)
1.2	Текущий контроль	Трансляция, компиляция и интерпретация – основные понятия. Формальные языки и грамматики.	ОПК-6.1 ОПК-6.2 ОПК-7.1	Доклад (устно)
1.3	Текущий контроль	Основные этапы компиляции. Языки высокого уровня, байт-код и ассемблеры	ОПК-6.1 ОПК-6.2 ОПК-7.1	Собеседование (устно)
1.4	Текущий контроль	Проектирование учебного компилятора	ОПК-6.1 ОПК-6.2 ОПК-7.1	Лабораторная работа (письменно/устно)
1.5	Текущий контроль	Проектирование учебного компилятора для курсовой работы	ОПК-6.1 ОПК-6.2 ОПК-7.1	Лабораторная работа (письменно/устно)
2.0	**Раздел 2. Методы, модели и алгоритмы лексического анализа**			
2.1	Текущий контроль	Проработка лекционного материала и подготовка к практическим занятиям раздела 2	ОПК-6.1 ОПК-6.2 ОПК-6.3	Конспект (письменно)
2.2	Текущий контроль	Задача лексического анализа. Регулярные языки, регулярные грамматики.	ОПК-6.1 ОПК-6.2 ОПК-6.3	Доклад (устно)
2.3	Текущий контроль	Конечные автоматы и регулярные выражения.	ОПК-6.1 ОПК-6.2 ОПК-6.3	Собеседование (устно)
2.4	Текущий контроль	Проектирование и разработка лексического анализатора учебного компилятора	ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
2.5	Текущий контроль	Проектирование и разработка лексического анализатора для курсовой работы	ОПК-6.1 ОПК-6.2 ОПК-6.3	Лабораторная работа (письменно/устно)
3.0	**Раздел 3. Методы, модели и алгоритмы синтаксического анализа**			
3.1	Текущий контроль	Проработка лекционного материала и подготовка к практическим занятиям раздела 3	ОПК-7.1 ОПК-7.2 ОПК-7.3	Конспект (письменно)
3.2	Текущий контроль	Задача синтаксического анализа. КС-грамматики.	ОПК-7.1 ОПК-7.2	Доклад (устно)

			ОПК-7.3	
3.3	Текущий контроль	Синтаксические распознаватели с возвратом.	ОПК-7.1 ОПК-7.2 ОПК-7.3	Собеседование (устно)
3.4	Текущий контроль	Нисходящие синтаксические распознаватели КС-языков без возврата	ОПК-7.1 ОПК-7.2 ОПК-7.3	Собеседование (устно)
3.5	Текущий контроль	Восходящие синтаксические распознаватели КС-языков без возврата	ОПК-7.1 ОПК-7.2 ОПК-7.3	Собеседование (устно)
3.6	Текущий контроль	Синтаксические распознаватели на основе грамматик предшествования	ОПК-7.1 ОПК-7.2 ОПК-7.3	Собеседование (устно)
3.7	Текущий контроль	Проектирование и разработка синтаксического анализатора на основе LL(1)-грамматики	ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
3.8	Текущий контроль	Проектирование и разработка синтаксического анализатора на основе грамматики операторного предшествования	ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
4.0	**Раздел 4. Методы, модели и алгоритмы семантического анализа. Синтаксически-управляемая трансляция**			
4.1	Текущий контроль	Проработка лекционного материала и подготовка к практическим занятиям раздела 4	ОПК-6.1 ОПК-7.2 ОПК-7.3	Конспект (письменно)
4.2	Текущий контроль	Методы, модели и алгоритмы семантического анализа. Граф потока управления и граф потоков данных.	ОПК-6.1 ОПК-7.2 ОПК-7.3	Доклад (устно)
4.3	Текущий контроль	Синтаксически-управляемая трансляция – понятия, возможности и алгоритмы применения	ОПК-6.1 ОПК-7.2 ОПК-7.3	Собеседование (устно)
4.4	Текущий контроль	Проектирование и разработка учебного семантического анализатора	ОПК-6.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
5.0	**Раздел 5. Методы и модели создания внутреннего представления программы. Современные подходы и критерии оптимизации программ при компиляции**			
5.1	Текущий контроль	Проработка лекционного материала и подготовка к практическим занятиям раздела 5	ОПК-7.1 ОПК-7.2 ОПК-7.3	Конспект (письменно)
5.2	Текущий контроль	Внутреннее представление программы – основные понятия, виды, алгоритмы генерации.	ОПК-7.1 ОПК-7.2 ОПК-7.3	Собеседование (устно)
5.3	Текущий контроль	Современные подходы и критерии оптимизации программ при компиляции.	ОПК-7.1 ОПК-7.2 ОПК-7.3	Собеседование (устно)
5.4	Текущий контроль	Разработка алгоритма генерации и оптимизации триад для учебного компилятора	ОПК-7.1 ОПК-7.2 ОПК-7.3	Собеседование (устно)
5.5	Текущий контроль	Проектирование и разработка подсистемы оптимизации триад для курсовой работы	ОПК-7.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
6.0	**Раздел 6. Методы и алгоритмы генерации объектного кода**			
6.1	Текущий контроль	Проработка лекционного материала и подготовка к практическим занятиям раздела 6	ОПК-6.1 ОПК-7.2 ОПК-7.3	Конспект (письменно)
6.2	Текущий контроль	Задача генерации объектного кода и способы ее решения.	ОПК-6.1 ОПК-7.2	Собеседование (устно)

			ОПК-7.3	
6.3	Текущий контроль	Проектирование и реализация модуля оптимизации внутреннего представления программы и генерации объектного кода	ОПК-6.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
6.4	Текущий контроль	Проектирование и разработка подсистемы кодогенерации для курсовой работы	ОПК-6.1 ОПК-7.2 ОПК-7.3	Лабораторная работа (письменно/устно)
	Промежуточная аттестация	Разделы: 1. Понятие трансляции, трансляция, компиляция и интерпретация. Основные этапы компиляции. 2. Методы, модели и алгоритмы лексического анализа 3. Методы, модели и алгоритмы синтаксического анализа 4. Методы, модели и алгоритмы семантического анализа. Синтаксически-управляемая трансляция. 5. Методы и модели создания внутреннего представления программы. Современные подходы и критерии оптимизации программ при компиляции 6. Методы и алгоритмы генерации объектного кода.	ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	Экзамен (собеседование) Экзамен - тестирование (компьютерные технологии)

*Форма проведения контрольно-оценочного мероприятия: устно, письменно, компьютерные технологии.

Описание показателей и критериев оценивания компетенций.
Описание шкал оценивания

Контроль качества освоения дисциплины включает в себя текущий контроль успеваемости и промежуточную аттестацию. Текущий контроль успеваемости и промежуточная аттестация обучающихся проводятся в целях установления соответствия достижений обучающихся поэтапным требованиям образовательной программы к результатам обучения и формирования компетенций.

Текущий контроль успеваемости – основной вид систематической проверки знаний, умений, навыков обучающихся. Задача текущего контроля – оперативное и регулярное управление учебной деятельностью обучающихся на основе обратной связи и корректировки. Результаты оценивания учитываются в виде средней оценки при проведении промежуточной аттестации.

Для оценивания результатов обучения используется четырехбалльная шкала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» и двухбалльная шкала: «зачтено», «не зачтено».

Перечень оценочных средств, используемых для оценивания компетенций, а также краткая характеристика этих средств приведены в таблице.

Текущий контроль

№	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Собеседование	Средство контроля на практическом занятии, организованное как специальная беседа преподавателя с	Вопросы для собеседования по

		обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п. Может быть использовано для оценки знаний обучающихся	темам/разделам дисциплины
2	Доклад	Продукт самостоятельной работы обучающегося, представляющий собой публичное выступление по представлению полученных результатов решения определенной учебно-практической, учебно-исследовательской или научной темы. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Темы докладов
3	Конспект	Особый вид текста, в основе которого лежит аналитико-синтетическая переработка информации первоисточника (исходного текста). Цель этой деятельности — выявление, систематизация и обобщение (с возможной критической оценкой) наиболее ценной (для конспектирующего) информации. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Темы конспектов
4	Лабораторная работа	Средство, позволяющее оценить умение обучающегося письменно/устно излагать суть поставленной задачи, самостоятельно применять стандартные методы решения поставленной задачи с использованием имеющейся лабораторной базы, проводить анализ полученного результата работы. Может быть использовано для оценки умений, навыков и (или) опыта деятельности обучающихся	Образец задания для выполнения лабораторной работы и примерный перечень вопросов для ее защиты

Промежуточная аттестация

№	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Экзамен	Средство, позволяющее оценить знания, умения, навыков и (или) опыта деятельности обучающегося по дисциплине. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Перечень теоретических вопросов и практических заданий (образец экзаменационного билета) к экзамену
2	Тест – промежуточная аттестация в форме экзамена	Система автоматизированного контроля освоения компетенций (части компетенций) обучающимся по дисциплине (модулю) с использованием информационно-коммуникационных технологий. Может быть использовано для оценки знаний, умений, навыков и (или) опыта деятельности обучающихся	Фонд тестовых заданий
3	Курсовая работа	Позволяет оценить умения обучающихся самостоятельно конструировать свои знания в процессе решения практических задач и проблем, ориентироваться в информационном пространстве и уровень сформированности аналитических, исследовательских навыков, навыков практического и творческого мышления. Может выполняться в индивидуальном порядке или группой обучающихся. Может быть использовано для оценки умений, навыков и (или) опыта деятельности обучающихся в предметной или межпредметной областях	Образец задания для выполнения курсовой работы и примерный перечень вопросов для ее защиты

Критерии и шкалы оценивания компетенций в результате изучения дисциплины

при проведении промежуточной аттестации в форме экзамена. Шкала оценивания уровня освоения компетенций

Шкала оценивания	Критерии оценивания	Уровень освоения компетенции
«отлично»	Обучающийся правильно ответил на теоретические вопросы. Показал отличные знания в рамках учебного материала. Правильно выполнил практические задания. Показал отличные умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Ответил на все дополнительные вопросы	Высокий
«хорошо»	Обучающийся с небольшими неточностями ответил на теоретические вопросы. Показал хорошие знания в рамках учебного материала. С небольшими неточностями выполнил практические задания. Показал хорошие умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Ответил на большинство дополнительных вопросов	Базовый
«удовлетворительно»	Обучающийся с существенными неточностями ответил на теоретические вопросы. Показал удовлетворительные знания в рамках учебного материала. С существенными неточностями выполнил практические задания. Показал удовлетворительные умения и владения навыками применения полученных знаний и умений при решении задач в рамках учебного материала. Допустил много неточностей при ответе на дополнительные вопросы	Минимальный
«неудовлетворительно»	Обучающийся при ответе на теоретические вопросы и при выполнении практических заданий продемонстрировал недостаточный уровень знаний и умений при решении задач в рамках учебного материала. При ответах на дополнительные вопросы было допущено множество неправильных ответов	Компетенция не сформирована

Тест – промежуточная аттестация в форме экзамена

Критерии оценивания	Шкала оценивания
Обучающийся верно ответил на 90 – 100 % тестовых заданий при прохождении тестирования	«отлично»
Обучающийся верно ответил на 80 – 89 % тестовых заданий при прохождении тестирования	«хорошо»
Обучающийся верно ответил на 70 – 79 % тестовых заданий при прохождении тестирования	«удовлетворительно»
Обучающийся верно ответил на 69 % и менее тестовых заданий при прохождении тестирования	«неудовлетворительно»

Курсовая работа

Шкала оценивания	Критерии оценивания
«отлично»	Содержание курсовой работы полностью соответствует заданию. Представлены результаты обзора литературных и иных источников. Структура курсовой работы логически и методически выдержана. Все выводы и предложения убедительно аргументированы. Оформление курсовой работы и полученные результаты полностью отвечают требованиям, изложенным в методических указаниях. При защите курсовой работы обучающийся правильно и уверенно отвечает на вопросы преподавателя, демонстрирует глубокое знание теоретического материала, способен аргументировать собственные утверждения и выводы
«хорошо»	Содержание курсовой работы полностью соответствует заданию. Представлены результаты обзора литературных и иных источников. Структура курсовой работы логически и методически выдержана. Большинство выводов и предложений аргументировано. Оформление курсовой работы и полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две несущественные ошибки в использовании терминов, в построенных диаграммах и схемах. Наличествует незначительное количество грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся правильно и

	уверенно отвечает на большинство вопросов преподавателя, демонстрирует хорошее знание теоретического материала, но не всегда способен аргументировать собственные утверждения и выводы. При наводящих вопросах преподавателя исправляет ошибки в ответе
«удовлетворительно»	Содержание курсовой работы частично не соответствует заданию. Результаты обзора литературных и иных источников представлены недостаточно полно. Есть нарушения в логике изложения материала. Аргументация выводов и предложений слабая или отсутствует. Имеются одно-два существенных отклонений от требований в оформлении курсовой работы. Полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две существенных ошибки в использовании терминов, в построенных диаграммах и схемах. Много грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся допускает грубые ошибки при ответах на вопросы преподавателя и /или не дал ответ более чем на 30% вопросов, демонстрирует слабое знание теоретического материала, в большинстве случаев не способен уверенно аргументировать собственные утверждения и выводы
«неудовлетворительно»	Содержание курсовой работы в целом не соответствует заданию. Имеются более двух существенных отклонений от требований в оформлении курсовой работы. Большое количество существенных ошибок по сути работы, много грамматических и стилистических ошибок и др. Полученные результаты не отвечают требованиям, изложенным в методических указаниях. При защите курсовой работы обучающийся демонстрирует слабое понимание программного материала. Курсовая работа не представлена преподавателю. Обучающийся не явился на защиту курсовой работы

Критерии и шкалы оценивания результатов обучения при проведении текущего контроля успеваемости

Собеседование

Шкалы оценивания		Критерии оценивания
«отлично»	«зачтено»	Глубокое и прочное усвоение программного материала. Полные, последовательные, грамотные и логически излагаемые ответы при видоизменении задания. Обучающийся свободно справляется с поставленными задачами, может обосновать принятые решения, демонстрирует владение разносторонними навыками и приемами выполнения практических работ
«хорошо»		Знание программного материала, грамотное изложение, без существенных неточностей в ответе на вопрос, правильное применение теоретических знаний, владение необходимыми навыками при выполнении практических задач
«удовлетворительно»		Обучающийся демонстрирует усвоение основного материала, при ответе допускаются неточности, при ответе недостаточно правильные формулировки, нарушение последовательности в изложении программного материала, затруднения в выполнении практических заданий Слабое знание программного материала, при ответе возникают ошибки, затруднения при выполнении практических работ
«неудовлетворительно»	«не зачтено»	Не было попытки выполнить задание

Доклад

Шкалы оценивания		Критерии оценивания
«отлично»	«зачтено»	Доклад создан с использованием компьютерных технологий (презентация PowerPoint, Flash–презентация, видео-презентация и др.) Использованы дополнительные источники информации. Содержание заданной темы раскрыто в полном объеме. Отражена структура доклада (вступление, основная часть, заключение, присутствуют выводы и примеры). Оформление работы. Оригинальность выполнения (работа сделана самостоятельно, представлена впервые)
«хорошо»		Доклад создан с использованием компьютерных технологий (презентация PowerPoint, Flash–презентация, видео-презентация и др.) Содержание доклада включает в себя информацию из основных источников (методическое пособие), дополнительные источники информации не использовались. Содержание заданной темы раскрыто

		не в полном объеме. Структура доклада сохранена (вступление, основная часть, заключение, присутствуют выводы и примеры)
«удовлетворительно»		Доклад сделан устно, без использования компьютерных технологий. Содержание доклада ограничено информацией только из методического пособия. Содержание заданной темы раскрыто не в полном объеме. Отсутствуют выводы и примеры. Оригинальность выполнения низкая
«неудовлетворительно»	«не зачтено»	Доклад сделан устно, без использования компьютерных технологий и других наглядных материалов. Содержание ограничено информацией только из методического пособия. Заданная тема доклада не раскрыта, основная мысль доклада не передана

Конспект

Шкалы оценивания		Критерии оценивания
«отлично»	«зачтено»	Конспект по теме выполнен в обозначенный преподавателем срок. Конспект выполнен обучающимся по заданной теме в полном объеме с соблюдением необходимой последовательности. Обучающийся работал полностью самостоятельно; раскрыл тему полностью и ответил на все вопросы преподавателя по конкретной теме конспекта. Конспект оформлен аккуратно, в наиболее оптимальной для фиксации результатов форме
«хорошо»		Конспект по теме выполнен в обозначенный преподавателем срок. Конспект выполнен обучающимся по заданной теме в полном объеме с соблюдением необходимой последовательности. Обучающийся работал полностью самостоятельно; раскрыл тему не полностью и ответил на часть вопросов преподавателя по конкретной теме конспекта. Конспект оформлен аккуратно, с незначительными исправлениями
«удовлетворительно»		Конспект по теме выполнен в обозначенный преподавателем срок. Конспект выполнен обучающимся по заданной теме в не полном объеме с частичным соблюдением необходимой последовательности. Обучающийся работал полностью самостоятельно; раскрыл тему не полностью и ответил на часть вопросов преподавателя по конкретной теме конспекта. Конспект оформлен не аккуратно
«неудовлетворительно»	«не зачтено»	Конспект по теме не выполнен в обозначенный преподавателем срок. Конспект выполнен обучающимся не по заданной теме в не полном объеме без соблюдения необходимой последовательности. Обучающийся работал не самостоятельно; не раскрыл тему и не ответил на вопросы преподавателя по конкретной теме конспекта. Конспект оформлен не аккуратно

Лабораторная работа

Шкалы оценивания		Критерии оценивания
«отлично»	«зачтено»	Лабораторная работа выполнена в обозначенный преподавателем срок, письменный отчет без замечаний. Лабораторная работа выполнена обучающимся в полном объеме с соблюдением необходимой последовательности. Обучающийся работал полностью самостоятельно; показал необходимые для проведения работы теоретические знания, практические умения и навыки. Работа (отчет) оформлена аккуратно, в наиболее оптимальной для фиксации результатов форме
«хорошо»		Лабораторная работа выполнена в обозначенный преподавателем срок, письменный отчет с небольшими недочетами. Лабораторная работа выполнена обучающимся в полном объеме и самостоятельно. Допущены отклонения от необходимой последовательности выполнения, не влияющие на правильность конечного результата. Работа показывает знание обучающимся основного теоретического материала и овладение умениями,

		необходимыми для самостоятельного выполнения работы. Допущены неточности и небрежность в оформлении результатов работы (отчета)
«удовлетворительно»		Лабораторная работа выполнена с задержкой, письменный отчет с недочетами. Лабораторная работа выполняется и оформляется обучающимся при посторонней помощи. На выполнение работы затрачивается много времени. Обучающийся показывает знания теоретического материала, но испытывает затруднение при самостоятельной работе с источниками знаний или приборами
«неудовлетворительно»	«не зачтено»	Лабораторная работа не выполнена, письменный отчет не представлен. Результаты, полученные обучающимся, не позволяют сделать правильных выводов и полностью расходятся с поставленной целью. Показывается плохое знание теоретического материала и отсутствие необходимых умений. Лабораторная работа не выполнена, у учащегося отсутствуют необходимые для проведения работы теоретические знания, практические умения и навыки

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности

3.1 Типовые контрольные задания для проведения собеседования

Контрольные варианты заданий выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец типовых вариантов заданий для проведения собеседований.

Образец типового варианта вопросов для проведения собеседования
«Основные этапы компиляции. Языки высокого уровня, байт-код и ассемблеры»

1. Какие основные этапы проходит программа в процессе компиляции?
2. Расскажите про каждый из перечисленных этапов компиляции — лексический анализ, синтаксический анализ, семантический анализ, генерация кода и оптимизация.
3. Что такое языки высокого уровня и какие из них вы знаете?
4. Что такое байт-код и для чего он используется?
5. В чем отличие ассемблера от высокоуровневых языков программирования?

Образец типового варианта вопросов для проведения собеседования
«Проектирование учебного компилятора»

1. Какие основные этапы проектирования учебного компилятора вы можете назвать?
2. Какой язык программирования вы предпочтете использовать для разработки учебного компилятора и почему?
3. Какие основные функции должен выполнять учебный компилятор?
4. Какими алгоритмами и структурами данных вы планируете пользоваться при разработке учебного компилятора?
5. Какие проблемы вы можете встретить при проектировании и разработке учебного компилятора и как планируете их решать?

Образец типового варианта вопросов для проведения собеседования
«Проектирование учебного компилятора для курсовой работы»

1. Какой опыт у вас есть в области проектирования компиляторов?
2. Расскажите о вашем понимании учебного компилятора и его основных компонентов.
3. Какие языки программирования вы используете для разработки компиляторов?
4. Какие алгоритмы и структуры данных вы применяли при проектировании

компиляторов?
5. Какие проблемы могут возникнуть в процессе разработки учебного компилятора и как вы их решаете?

Образец типового варианта вопросов для проведения собеседования
«Конечные автоматы и регулярные выражения.»

1. Что такое конечный автомат?
2. Какие основные компоненты составляют конечный автомат?
3. Каковы основные типы конечных автоматов?
4. Чем отличается детерминированный и недетерминированный конечный автомат?
5. Какие операции можно выполнять с конечными автоматами?

Образец типового варианта вопросов для проведения собеседования
«Проектирование и разработка лексического анализатора учебного компилятора»

1. Какие навыки и знания нужны для разработки лексического анализатора учебного компилятора?
2. Как вы оцениваете свои знания в области проектирования и разработки компиляторов?
3. Какие языки программирования и инструменты вы бы использовали при разработке лексического анализатора?
4. Расскажите о том, какие шаги вы предпринимаете при проектировании лексического анализатора.
5. Какие алгоритмы лексического анализа вы знаете и какие из них вы бы использовали в данном проекте?

Образец типового варианта вопросов для проведения собеседования
«Проектирование и разработка лексического анализатора для курсовой работы»

1. Какие знания и навыки в области лексического анализа вы имеете?
2. Расскажите о вашем опыте работы с проектированием и разработкой лексических анализаторов.
3. Какие инструменты и технологии вы предпочитаете использовать при разработке лексического анализатора?
4. Каковы основные шаги проектирования и разработки лексического анализатора?
5. Какие проблемы и сложности вы можете встретить при разработке лексического анализатора?

Образец типового варианта вопросов для проведения собеседования
«Синтаксические распознаватели с возвратом.»

1. Какие основные принципы работы синтаксических распознавателей с возвратом?
2. Какие преимущества и недостатки имеют синтаксические распознаватели с возвратом по сравнению с другими методами синтаксического анализа?
3. Какие приложения могут использовать синтаксические распознаватели с возвратом?
4. Какие алгоритмы используются для реализации синтаксических распознавателей с возвратом?
5. Какие методы оптимизации можно применить для ускорения работы синтаксического распознавателя с возвратом?

Образец типового варианта вопросов для проведения собеседования
«Нисходящие синтаксические распознаватели КС-языков без возврата»

1. Что такое нисходящий синтаксический распознаватель КС-языков без возврата?

2. Какие основные принципы работы нисходящего синтаксического распознавателя КС-языков без возврата?
3. Какие преимущества и недостатки имеет использование нисходящих синтаксических распознавателей КС-языков без возврата?
4. Какие алгоритмы используются для реализации нисходящих синтаксических распознавателей КС-языков без возврата?
5. Какие типы ошибок исключаются при использовании нисходящих синтаксических распознавателей КС-языков без возврата?

Образец типового варианта вопросов для проведения собеседования
«Восходящие синтаксические распознаватели КС-языков без возврата»

1. Какие преимущества имеет восходящий синтаксический распознаватель перед другими методами распознавания КС-языков?
2. Что такое возврат в контексте синтаксического анализа и почему он необходим при использовании нисходящих методов?
3. В чем состоит основная идея восходящего синтаксического анализа без возврата?
4. Какие ограничения есть у восходящих синтаксических распознавателей без возврата, и как они могут влиять на эффективность и надежность анализа?

Образец типового варианта вопросов для проведения собеседования
«Синтаксические распознаватели на основе грамматик предшествования»

1. Что такое синтаксический распознаватель и зачем он нужен?
2. Чем отличаются синтаксические распознаватели на основе грамматик предшествования от других типов синтаксических анализаторов?
3. Какие основные компоненты составляют синтаксический распознаватель на основе грамматик предшествования?
4. Какие типы ошибок может обнаруживать такой распознаватель?
5. Какие преимущества и недостатки имеют синтаксические распознаватели на основе грамматик предшествования по сравнению с другими алгоритмами синтаксического анализа?

Образец типового варианта вопросов для проведения собеседования
«Проектирование и разработка синтаксического анализатора на основе LL(1)-грамматики»

1. Какие основные шаги необходимо выполнить для проектирования и разработки синтаксического анализатора на основе LL(1)-грамматики?
2. Какие преимущества имеет использование LL(1)-грамматики при разработке синтаксического анализатора?
3. Какие ограничения и ограничения существуют при использовании LL(1)-грамматики?

Образец типового варианта вопросов для проведения собеседования
«Проектирование и разработка синтаксического анализатора на основе грамматики операторного предшествования»

1. Расскажите о вашем опыте в проектировании и разработке синтаксических анализаторов.
2. Что такое грамматика операторного предшествования и какой ее роль при разработке синтаксического анализатора?
3. Какие основные шаги вы предпринимаете для проектирования и разработки синтаксического анализатора на основе грамматики операторного предшествования?

Образец типового варианта вопросов для проведения собеседования

«Синтаксически-управляемая трансляция – понятия, возможности и алгоритмы применения»

1. Что такое синтаксически-управляемая трансляция?
2. Какие возможности предоставляет синтаксически-управляемая трансляция?
3. Какие алгоритмы можно использовать при синтаксически-управляемой трансляции?

Образец типового варианта вопросов для проведения собеседования
«Проектирование и разработка учебного семантического анализатора»

1. Какова ваша экспертиза в области проектирования и разработки учебных семантических анализаторов?
2. Можете ли вы рассказать нам о проектах, в которых вы принимали участие по проектированию и разработке учебных семантических анализаторов? Какова была ваша роль в этих проектах?
3. Какие инструменты и технологии вы обычно используете при проектировании и разработке учебных семантических анализаторов?

Образец типового варианта вопросов для проведения собеседования
«Внутреннее представление программы – основные понятия, виды, алгоритмы генерации.»

1. Как бы вы описали внутреннее представление программы?
2. Какие основные понятия связаны с внутренним представлением программы?
3. Расскажите о различных видах внутреннего представления программы.

Образец типового варианта вопросов для проведения собеседования
«Современные подходы и критерии оптимизации программ при компиляции.»

1. Какие подходы к оптимизации программ при компиляции вы знаете?
2. Какие критерии оптимизации программа должна удовлетворять?
3. Какими методами можно проводить оптимизацию при компиляции?

Образец типового варианта вопросов для проведения собеседования
«Разработка алгоритма генерации и оптимизации триад для учебного компилятора»

1. Какие языки программирования вы знаете и какой опыт использования у вас есть?
2. Что такое триада в контексте компиляторов?
3. Как вы будете оценивать эффективность генерируемых триад в вашем компиляторе?

Образец типового варианта вопросов для проведения собеседования
«Проектирование и разработка подсистемы оптимизации триад для курсовой работы»

1. Что вы понимаете под оптимизацией триад?
2. Какие языки программирования и технологии вы использовали для разработки подобных проектов?
3. Расскажите о вашем опыте работы с алгоритмами оптимизации. Какие алгоритмы вы применяли раньше?

Образец типового варианта вопросов для проведения собеседования
«Задача генерации объектного кода и способы ее решения.»

1. Расскажите, как происходит процесс генерации объектного кода?
2. Какие компоненты программы могут быть преобразованы в объектный код?
3. Какие способы существуют для генерации объектного кода?
4. Каким образом осуществляется оптимизация кода перед его генерацией в объектный

код?

Образец типового варианта вопросов для проведения собеседования
«Проектирование и реализация модуля оптимизации внутреннего представления программы и генерации объектного кода»

1. Какие инструменты и технологии вы использовали для проектирования и реализации подобного модуля?
2. Какие методы и алгоритмы оптимизации программного кода вы применяли в прошлых проектах?
3. Какие показатели эффективности получены в результате применения оптимизаций внутреннего представления программы и генерации объектного кода?

Образец типового варианта вопросов для проведения собеседования
«Проектирование и разработка подсистемы кодогенерации для курсовой работы»

1. Расскажите о вашем опыте в проектировании и разработке программного обеспечения?
2. Какие технологии и инструменты вы использовали при разработке подсистемы кодогенерации?
3. Какие языки программирования вы знаете и какой из них вы считаете наиболее подходящим для реализации данной подсистемы?

3.2 Типовые контрольные темы для написания докладов

Контрольные варианты заданий выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец типовых вариантов тем для написания докладов.

Образец тем докладов
«Трансляция, компиляция и интерпретация – основные понятия. Формальные языки и грамматики.»

1. Трансляция, компиляция и интерпретация: сравнение и особенности каждого подхода.
2. Роль формальных языков и грамматик в компиляции и интерпретации программ.
3. Синтаксический анализ и построение дерева разбора в формальных языках.

Образец тем докладов
«Задача лексического анализа. Регулярные языки, регулярные грамматики.»

1. Определения и примеры регулярных множеств
2. Конечные автоматы и их использование в теории вычислений
3. Автоматы с магазинной памятью: особенности, примеры, применение.

Образец тем докладов
«Задача синтаксического анализа. КС-грамматики.»
Определение LL(1)-грамматик и LR(1)-таблиц разбора

1. Реализация LL и LR методов разбора
2. Сравнение LL и LR методов разбора: преимущества, недостатки, области применения.

Образец тем докладов
«Методы, модели и алгоритмы семантического анализа. Граф потока управления и граф потоков данных.»

1. Основные методы и модели семантического анализа текстового контента.

2. Применение алгоритмов машинного обучения в семантическом анализе текста.
3. Анализ графа потока управления в программировании.

3.3 Типовые контрольные задания для написания конспекта

Контрольные варианты заданий выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец типовых вариантов заданий для написания конспектов.

Образец тем конспектов
«Проработка лекционного материала и подготовка к практическим занятиям раздела 1»

1. Определение и классификация языков программирования.
2. Исторический контекст развития языков программирования.
3. Введение в концепцию высокоуровневых и низкоуровневых языков программирования.
4. Способы передачи инструкций компьютеру с помощью языка программирования.
5. Понятие спецификации в контексте языка программирования.
6. Обзор и анализ различных языков программирования.
7. Роль и влияние языков программирования на современные технологии.

Образец тем конспектов
«Проработка лекционного материала и подготовка к практическим занятиям раздела 2»

1. Введение в понятие эффективности языка программирования.
2. Методы и инструменты оценки эффективности языка программирования.
3. Анализ критериев эффективности: время выполнения, обучение, ресурсы, продуктивность разработчика и т.д.
4. Рассмотрение влияния выбора языка программирования на результаты проекта.
5. Области применения различных языков программирования.
6. Подробный анализ нескольких примеров использования конкретных языков программирования.
7. Изучение современных тенденций в развитии языков программирования и их перспектив.

Образец тем конспектов
«Проработка лекционного материала и подготовка к практическим занятиям раздела 3»

1. Определение выражений и присваивания в контексте программирования.
2. Анализ различных видов выражений: арифметических, логических, условных и т.д.
3. Обсуждение префиксной, постфиксной и инфиксной форм записи. Примеры и сравнение.
4. Введение в операции в программах: арифметические, сравнения, логические и т.д.
5. Описание процесса оценки и выполнения выражений и операций.
6. Обсуждение влияния порядка операций на результат вычисления выражений.
7. Применение полученных знаний на практических примерах.

Образец тем конспектов
«Проработка лекционного материала и подготовка к практическим занятиям раздела 4»

1. Определение синтаксиса и семантики в контексте языков программирования.
2. Анализ основных синтаксических элементов: ключевых слов, операторов, идентификаторов и т.д.
3. Рассмотрение важности правильного синтаксиса для успешного выполнения программ.
4. Введение в понятие семантики и его значение для понимания и анализа программ.
5. Обсуждение различия между статической и динамической семантикой.

6. Рассмотрение влияния семантики на процесс отладки и тестирования программ.
7. Применение полученных знаний на практических примерах.

Образец тем конспектов
«Проработка лекционного материала и подготовка к практическим занятиям раздела 5»

1. Понимание типизации данных и её важности в программировании.
2. Обсуждение примитивных типов данных, включая числа, булевы значения и символы.
3. Введение в составные типы данных, включая массивы, структуры и объединения.
4. Обзор указателей и их важности для работы с памятью и структурами данных.
5. Обсуждение статической и динамической типизации, их преимуществ и недостатков.
6. Обзор систем типов в разных языках программирования.
7. Применение полученных знаний на практических примерах.

Образец тем конспектов
«Проработка лекционного материала и подготовка к практическим занятиям раздела 6»

1. Введение в LL(1)-грамматики и их использование в синтаксическом анализе.
2. Обсуждение LR(1) - таблиц разбора и их применения в программировании.
3. Рассмотрение различий между LL и LR методами разбора.
4. Обзор примеров применения этих методов на практике.
5. Разбор основных проблем, которые могут возникнуть при использовании каждого из методов.
6. Рассмотрение того, как эти методы влияют на производительность и управляемость кода.
7. Применение полученных знаний на практических примерах.

3.4 Типовые задания для выполнения лабораторной работы и примерный перечень вопросов для ее защиты

Контрольные варианты заданий выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец типовых вариантов заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты.

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Проектирование учебного компилятора»

Составить автоматную грамматику и на ее основе реализовать лексический анализатор языка, цепочки которого имеют вид, указанный в задании. Лексический анализатор должен преобразовывать исходный текст в последовательность лексем. По результатам работы анализатора должны формироваться таблицы идентификаторов и констант.

Контрольные вопросы:

1. Назовите основные этапы компиляции.
2. Дайте краткую характеристику этапу лексического анализа.
3. Что такое лексема? Приведите примеры.
4. С какими таблицами осуществляется работа на этапе лексического анализа?
5. Дайте определение грамматики
6. Дайте определение автоматной грамматики
7. Назовите основные этапы создания анализатора по автоматной грамматике.
8. Какие способы задания автоматного языка Вы знаете?
9. Дайте определение конечного автомата.
10. Что такое детерминированная, недетерминированная и вполне детерминированная

формы автоматных грамматик? Приведите примеры.

11. Дайте определение детерминированному (ДКА) и недетерминированному конечному автомату (НКА).
12. В чем заключается алгоритм перехода от НКА к ДКА. Приведите пример.
13. Как перейти от грамматики в детерминированной форме к грамматике во вполне детерминированной форме?
14. Как по вполне детерминированной автоматной грамматике составить программу

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Проектирование учебного компилятора для курсовой работы»

Построить простой лексический анализатор; найденные лексемы выделить.

Контрольные вопросы:

1. Объясните основные конструкции, предусмотренные в грамматике разработанного языка.
2. Какие лексические единицы необходимо выделять при лексическом анализе и каким образом это осуществляется?
3. Какие структуры данных использовались для представления абстрактного синтаксического дерева?
4. Как осуществляется интерпретация программы в разработанном языке? Какие операции поддерживает интерпретатор?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Проектирование и разработка лексического анализатора учебного компилятора»

Составить автоматную грамматику и на ее основе реализовать лексический анализатор языка, цепочки которого имеют вид, указанный в задании. Лексический анализатор должен преобразовывать исходный текст в последовательность лексем. По результатам работы анализатора должны формироваться таблицы идентификаторов и констант.

Контрольные вопросы:

1. Назовите основные этапы компиляции.
2. Дайте краткую характеристику этапу лексического анализа.
3. Что такое лексема? Приведите примеры.
4. С какими таблицами осуществляется работа на этапе лексического анализа?
5. Дайте определение грамматики
6. Дайте определение автоматной грамматики
7. Назовите основные этапы создания анализатора по автоматной грамматике.
8. Какие способы задания автоматного языка Вы знаете?
9. Дайте определение конечного автомата.
10. Что такое детерминированная, недетерминированная и вполне детерминированная формы автоматных грамматик? Приведите примеры.
11. Дайте определение детерминированному (ДКА) и недетерминированному конечному автомату (НКА).
12. В чем заключается алгоритм перехода от НКА к ДКА. Приведите пример.
13. Как перейти от грамматики в детерминированной форме к грамматике во вполне детерминированной форме?
14. Как по вполне детерминированной автоматной грамматике составить программу

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Проектирование и разработка лексического анализатора для курсовой работы»

Составить автоматную грамматику и на ее основе реализовать лексический анализатор языка, цепочки которого имеют вид, указанный в задании. Лексический анализатор должен преобразовывать исходный текст в последовательность лексем. По результатам работы анализатора должны формироваться таблицы идентификаторов и констант.

Контрольные вопросы:

15. Назовите основные этапы компиляции.
16. Дайте краткую характеристику этапу лексического анализа.
17. Что такое лексема? Приведите примеры.
18. С какими таблицами осуществляется работа на этапе лексического анализа?
19. Дайте определение грамматики
20. Дайте определение автоматной грамматики
21. Назовите основные этапы создания анализатора по автоматной грамматике.
22. Какие способы задания автоматного языка Вы знаете?
23. Дайте определение конечного автомата.
24. Что такое детерминированная, недетерминированная и вполне детерминированная формы автоматных грамматик? Приведите примеры.
25. Дайте определение детерминированному (ДКА) и недетерминированному конечному автомату (НКА).
26. В чем заключается алгоритм перехода от НКА к ДКА. Приведите пример.
27. Как перейти от грамматики в детерминированной форме к грамматике во вполне детерминированной форме?
28. Как по вполне детерминированной автоматной грамматике составить программу

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты

«Проектирование и разработка синтаксического анализатора на основе LL(1)-грамматики»

С использованием метода рекурсивного спуска реализовать синтаксический анализатор языка, цепочки которого имеют вид, указанный в задании на первую лабораторную работу. Синтаксический анализатор должен принимать на вход последовательность лексем, сформированную лексическим анализатором и восстанавливать дерево разбора цепочки. В случае ошибок во входной цепочке анализатор должен дать пользователю информативное сообщение с указанием причины и места возникновения ошибки.

Контрольные вопросы:

1. Дайте определение контекстно-свободной грамматики
2. Скажите, какой класс грамматик допускает построение анализаторов методом
3. Как строится анализатор методом рекурсивного спуска?
4. Какие элементы используются в расширенной бэкусовой нормальной форме?
5. Что такое LL(1)-грамматика и как она используется при разработке синтаксического анализатора?
6. Какие основные шаги включает процесс разработки синтаксического анализатора на основе LL(1)-грамматики?
7. Чем отличается лексический анализатор от синтаксического анализатора?
8. Какие тесты можно провести для проверки корректной работы лексического анализатора?
9. Какие ошибки могут возникнуть при работе синтаксического анализатора и как их обнаружить?
10. Какие методы оптимизации можно применить для улучшения работы синтаксическим анализатором на основе LL(1)-грамматики?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Проектирование и разработка синтаксического анализатора на основе грамматики операторного предшествования»

Разработка грамматики операторного предшествования для заданного языка программирования.

Контрольные вопросы:

1. Что такое синтаксический анализ и какие задачи он решает в процессе компиляции программ?
2. Что такое грамматика операторного предшествования и как она строится?
3. Какие особенности и преимущества имеет синтаксический анализ на основе грамматики операторного предшествования?
4. Какие этапы включает процесс разработки синтаксического анализатора на основе грамматики операторного предшествования?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Проектирование и разработка учебного семантического анализатора»

Дополнить анализатор, разработанный в рамках лабораторных работ №1 и №2, этапом формирования внутренней формы представления программы.

Контрольные вопросы:

1. Что такое внутренняя форма представления программы? Зачем она используется?
2. Какие формы внутреннего представления программ Вы знаете?
3. Что такое трехадресный код? Приведите пример.
4. В чем отличие триад и тетрад? Приведите пример.
5. В чем особенности постфиксной формы записи?
6. Как вычислить выражение в ПОЛИЗе вручную?
7. Как перевести выражение в ПОЛИЗ вручную?
8. Как представить в ПОЛИЗе операции присваивания и обращения по индексу?
9. Как представить в ПОЛИЗе условный оператор?
10. Как представить в ПОЛИЗе операторы цикла?
11. Существуют ли преимущества у постфиксной формы записи перед традиционной?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Проектирование и разработка подсистемы оптимизации триад для курсовой работы»

Проектирование и разработка подсистемы оптимизации триад для курсовой работы.

Разработать структуру и архитектуру подсистемы оптимизации триад, определить основные компоненты и их взаимодействие. Разработать модуль оптимизации триады, который выполняет оптимизацию на основе выбранного критерия.

Контрольные вопросы:

1. Что такое триада и почему ее оптимизация важна?
2. Какие критерии и ограничения могут быть применены при оптимизации триады?
3. Какие компоненты входят в подсистему оптимизации триад и как они взаимодействуют между собой?
4. Какой алгоритм оптимизации был выбран для подсистемы и почему?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для

их защиты
«Проектирование и реализация модуля оптимизации внутреннего представления программы и генерации объектного кода»

Разработать модуль оптимизации внутреннего представления программы и генерации объектного кода для заданного языка программирования. Разработать алгоритмы генерации объектного кода, включая:

- Построение графа потока управления
- Аллокацию регистров
- Преобразование в ассемблерный код

Контрольные вопросы:

1. Какую роль играют модули оптимизации и генерации объектного кода в процессе компиляции программы?
2. Какие алгоритмы оптимизации вы использовали в вашем модуле? Опишите их принцип работы.
3. Какие алгоритмы генерации объектного кода вы реализовали? Каким образом они преобразуют внутреннее представление программы в исполняемый код?

Образец заданий для выполнения лабораторных работ и примерный перечень вопросов для их защиты
«Проектирование и разработка подсистемы кодогенерации для курсовой работы»

Разработка компонентов подсистемы кодогенерации, основанных на выбранной архитектуре. Тестирование разработанных компонентов и их интеграция в общую систему.

Контрольные вопросы:

1. Какие подходы к кодогенерации существуют?
2. Какие требования были поставлены перед разрабатываемой подсистемой кодогенерации?
3. Почему была выбрана данная архитектура для подсистемы кодогенерации?
4. Какие компоненты были разработаны для подсистемы кодогенерации?

3.5 Типовые контрольные задания для проведения тестирования

Фонд тестовых заданий по дисциплине содержит тестовые задания, распределенные по разделам и темам, с указанием их количества и типа.

Структура фонда тестовых заданий по дисциплине

Индикатор достижения компетенции	Тема в соответствии с РПД	Характеристика ТЗ	Количество тестовых заданий, типы ТЗ
ОПК-6.1 ОПК-6.2 ОПК-7.1	Проработка лекционного материала и подготовка к практическим занятиям раздела 1	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-6.2 ОПК-7.1	Трансляция, компиляция и интерпретация – основные понятия. Формальные языки и грамматики.	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ

ОПК-6.1 ОПК-6.2 ОПК-7.1	Основные этапы компиляции. Языки высокого уровня, байт-код и ассемблеры	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-6.2 ОПК-7.1	Проектирование учебного компилятора	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-6.2 ОПК-7.1	Проектирование учебного компилятора для курсовой работы	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-6.2 ОПК-6.3	Проработка лекционного материала и подготовка к практическим занятиям раздела 2	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-6.2 ОПК-6.3	Задача лексического анализа. Регулярные языки, регулярные грамматики.	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-6.2 ОПК-6.3	Конечные автоматы и регулярные выражения.	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-6.2 ОПК-6.3	Проектирование и разработка лексического анализатора учебного компилятора	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-6.2 ОПК-6.3	Проектирование и разработка лексического анализатора для курсовой работы	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Проработка лекционного материала и подготовка к практическим занятиям раздела 3	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1	Задача синтаксического анализа. КС-грамматики.	Знание	1 – ОТЗ

ОПК-7.2 ОПК-7.3			1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Синтаксические распознаватели с возвратом.	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Нисходящие синтаксические распознаватели КС-языков без возврата	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Восходящие синтаксические распознаватели КС-языков без возврата	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Синтаксические распознаватели на основе грамматик предшествования	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Проектирование и разработка синтаксического анализатора на основе LL(1)-грамматики	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Проектирование и разработка синтаксического анализатора на основе грамматики операторного предшествования	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-7.2 ОПК-7.3	Проработка лекционного материала и подготовка к практическим занятиям раздела 4	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-7.2 ОПК-7.3	Методы, модели и алгоритмы семантического анализа. Граф потока управления и граф потоков данных.	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-7.2	Синтаксически-управляемая трансляция – понятия, возможности и алгоритмы применения	Знание	1 – ОТЗ 1 – ЗТЗ

ОПК-7.3		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-7.2 ОПК-7.3	Проектирование и разработка учебного семантического анализатора	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Проработка лекционного материала и подготовка к практическим занятиям раздела 5	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Внутреннее представление программы – основные понятия, виды, алгоритмы генерации.	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Современные подходы и критерии оптимизации программ при компиляции.	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Разработка алгоритма генерации и оптимизации триад для учебного компилятора	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-7.1 ОПК-7.2 ОПК-7.3	Проектирование и разработка подсистемы оптимизации триад для курсовой работы	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-7.2 ОПК-7.3	Проработка лекционного материала и подготовка к практическим занятиям раздела 6	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-7.2 ОПК-7.3	Задача генерации объектного кода и способы ее решения.	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-7.2 ОПК-7.3	Проектирование и реализация модуля оптимизации внутреннего представления программы и генерации объектного кода	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ

			1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
ОПК-6.1 ОПК-7.2 ОПК-7.3	Проектирование и разработка подсистемы кодогенерации для курсовой работы	Знание	1 – ОТЗ 1 – ЗТЗ
		Умение	1 – ОТЗ 1 – ЗТЗ
		Навык и (или) опыт деятельности/ действие	1 – ОТЗ 1 – ЗТЗ
		Итого	93 – ОТЗ 93 – ЗТЗ

Полный комплект ФТЗ хранится в электронной информационно-образовательной среде ИрГУПС и обучающийся имеет возможность ознакомиться с демонстрационным вариантом ФТЗ.

Ниже приведен образец типового варианта итогового теста, предусмотренного рабочей программой дисциплины.

1. Какой формат записи операций используется в языках программирования большинством?

   a) Префиксная
   b) Постфиксная
   c) Инфиксная
   d) Нет фиксированного формата

2. Что из перечисленного относится к семантике языка программирования?

   a) Структура кода
   b) Способ записи операций
   c) Смысловое значение конструкций кода
   d) Формат комментариев

3. Какой из следующих типов данных не считается примитивным?

   a) int
   b) char
   c) float
   d) Array

4. Что означает передача параметров "по значению"?

   a) Передается ссылка на переменную
   b) Передается копия значения переменной
   c) Передается адрес переменной в памяти
   d) Передается последнее использованное значение переменной

5. Что такое метод в контексте объектно-ориентированного программирования?

   a) Отдельный модуль программы
   b) Функция, определенная внутри класса
   c) Объект класса
   d) Тип данных

6. Что из нижеперечисленного является примером регулярного множества?

   a) **Множество всех слов над алфавитом {a, b}, в которых не встречаются две буквы "a" подряд**
   b) Множество всех целых чисел
   c) Множество всех реальных чисел
   d) Множество всех точек на плоскости

7. Какие грамматики используются для создания LL(1)-таблицы разбора?

   a) **Грамматики, для которых можно построить дерево разбора без возвратов**
   b) Грамматики, содержащие левую рекурсию
   c) Грамматики, содержащие правую рекурсию
   d) Грамматики, содержащие две одинаковые правила продукции

8. Чем отличаются LL и LR методы разбора?

   a) Видом используемой грамматики
   b) Порядком чтения входного потока
   c) Видом дерева разбора
   d) Методами обработки ошибок

9. Что такое конечный автомат?

   a) Устройство для выполнения арифметических операций
   b) Модель вычислений с ограниченным числом состояний
   c) Модель хранения данных
   d) Специализированный тип микропроцессора

10. Что такое язык программирования, и как он используется для создания сообщений для компьютера?

Ответ: Язык программирования - это стандартизированный коммуникационный инструмент, который используется для выражения инструкций, которые могут быть выполнены компьютером. Он позволяет программистам писать программы, которые в конечном итоге преобразуются в машинный код, понятный компьютеру.

11. Какие основные критерии используются для оценки эффективности языков программирования?

Ответ: Эффективность языка программирования часто оценивается по его производительности (скорости выполнения кода), понятности (легкости чтения и понимания кода), универсальности (способности решать различные задачи), наличию поддержки и сообщества, и т.д.

12. Что такое префиксная, постфиксная и инфиксная формы записи в контексте выражений и присваиваний в языках программирования?

Ответ: Префиксная, постфиксная и инфиксная формы записи связаны с размещением операторов относительно операндов в выражениях. Префиксная запись (или обратная польская запись) предполагает размещение оператора перед операндами (например, "+ a b"). Постфиксная (или польская запись) размещает оператор после операндов (например, "a b +"). Инфиксная запись, которая является наиболее распространенной в большинстве языков программирования, размещает оператор между операндами (например, "a + b").

13. В чем состоит различие между синтаксисом и семантикой языка программирования?

Ответ: Синтаксис в контексте языков программирования относится к правилам, которые определяют, какие комбинации символов могут считаться правильно структурированными программами или фрагментами программ на данном языке. Семантика языка программирования относится к значению или поведению этих программ или фрагментов программ.

14. Что такое типизация данных и какие существуют примитивные и составные типы?

Ответ: Типизация данных в языках программирования относится к системе, которая позволяет программам и компиляторам понимать, какой тип данных хранится в определенной переменной, что в свою очередь определяет, какие операции могут быть выполнены с этими данными. Примитивные типы данных - это наиболее базовые типы данных, которые предоставляют основу для построения более сложных типов данных. Они включают в себя типы, такие как integer, float, boolean, char и т.д. Составные типы данных, такие как массивы, структуры, классы, это типы данных, которые могут содержать значения или переменные более примитивных типов или других составных типов.

15. Как работают подпрограммы и какие существуют способы передачи параметров?

Ответ: Подпрограммы (также известные как функции или процедуры) это сегменты кода, которые можно вызывать из любой части программы. Они могут принимать параметры и возвращать результат. Параметры могут передаваться по значению (где используется копия значения) или по ссылке (где используется сама переменная).

16. Что такое объектно-ориентированный язык программирования и каковы его основные характеристики?

Ответ: Объектно-ориентированный язык программирования - это язык, который поддерживает или использует парадигму программирования, основанную на "объектах", которые могут содержать данные и методы для работы с этими данными. Основные характеристики включают инкапсуляцию, наследование и полиморфизм.

17. Что такое регулярные множества и конечные автоматы?

Ответ: Регулярные множества - это тип языка в теории формальных языков, который может быть распознан конечным автоматом. Конечный автомат - это абстрактная машина, которая может быть в ограниченном числе состояний. Он может менять свое состояние в ответ на входные данные, следуя определенному набору переходов.

18. Что такое LL(1) и LR(1) грамматики и каковы основные различия между LL и LR методами разбора?

Ответ: LL(1) и LR(1) - это классы грамматик в теории формальных языков, используемых для разбора языков программирования. Они отличаются по своему подходу к анализу: LL(1) анализирует ввод слева направо и строит левостороннюю разбивку, тогда как LR(1) также анализирует ввод слева направо, но строит правостороннюю разбивку. LR грамматики более мощные, чем LL, и могут анализировать более широкий класс языков.

3.6 Типовое задание для выполнения курсовой работы

Типовые задания выложены в электронной информационно-образовательной среде ИрГУПС, доступной обучающемуся через его личный кабинет.

Ниже приведен образец задания для выполнения курсовой работы и примерный перечень вопросов для ее защиты.

Образец типового задания для выполнения курсовой работы

Курсовая работа заключается в создании компилятора с заданного подмножества языка (Паскаль, Delphi, Си, Си++, Си#, Java) с незначительными модификациями и упрощениями (полное описание входного и выходного языков дано далее в задании для каждого варианта). Результатами курсовой работы являются программная реализация заданного компилятора и пояснительная записка, оформленная в соответствии с требованиями нормоконтроля.
Компилятор рекомендуется построить из следующих составных частей:

1. лексический анализатор;
2. синтаксический анализатор;
3. оптимизатор;
4. генератор результирующего кода.

Для построения компилятора рекомендуется использовать методы, освоенные в ходе выполнения лабораторных работ по курсу «Теория языков программирования и методы трансляции», а также в процессе самостоятельного изучения теоретического материала.

Образец типовых вопросов для защиты курсовых работ

1. Дайте определения понятия транслятор, компилятор, интерпретатор.
2. В чем преимущества и недостатки интерпретаторов?
3. Какие внутренние формы представления могут использоваться при интерпретации?
5. Что такое динамическая компиляция?
6. В чем заключается алгоритм интерпретации польской инверсной записи с
7. Расскажите об особенностях реализации команд при создании интерпретатора.

3.7 Перечень теоретических вопросов к экзамену

(для оценки знаний)

1. Цепочки символов и операции над ними. Необходимые понятия.
2. Понятие языка. Необходимые определения (конкатенация, эквивалентность, замыкание Клини…). Способы задания языков; возможные проблемы.
3. Особенности языков программирования с точки зрения теории формальных языков.
4. Понятие и формальное определение грамматики как способа задания языка. Описание языка программирования посредством грамматик.
5. Способы задания грамматик (форма Бэкуса-Наура, запись с использованием метасимволов).
6. Классификация грамматик и языков по Хомскому.
7. Вывод и выводимость цепочек символов. Понятие сентенциальной формы. Деревья вывода цепочек языка, алгоритмы их построения.
8. Распознаватели формальных языков – структура, необходимые понятия (такт, конфигурация, допустимость цепочек).
9. Виды распознавателей формальных языков и их классификация. Задача разбора в теории формальных языков.
10. Понятие трансляции, компиляции и интерпретации. Транслятора, компиляторы, интерпретаторы – понятия, сходства и различия.
11. Общая схема работы компилятора. Этапы компиляции.
12. Назначение и особенности построения таблиц идентификаторов.
13. Простейшие методы построения таблиц идентификаторов.
14. Построение таблиц идентификаторов по методу бинарного дерева.
15. Хеширование. Построение таблиц идентификаторов с помощью хеш-адресации.

16. Лексический анализатор (сканер) – назначение, способы построения, проблемы построения.
17. Регулярные грамматики – определение, способы задания, разновидности регулярных грамматик.
18. Регулярные множества и регулярные выражения – определение, свойства.
19. Автоматные грамматики – определение, сущность и способ приведения регулярной грамматики к автоматному виду.
20. Конечный автомат (КА) как формальная конструкция, задающая язык. Способы задания КА, необходимые понятия (такт, конфигурация, ДКА, НКА, полностью определенный КА).
21. Построение ДКА, эквивалентного заданному НКА. Классы эквивалентности, минимизация ДКА.
22. Эквивалентные способы задания регулярных языков.
23. Синтаксический анализатор (парсер) – назначение, способы построения, проблемы построения.
24. Автоматы с магазинной памятью как распознаватели КС-языков; необходимые определения (такт, конфигурация, функция перехода), классификация автоматов.
25. Виды и цели преобразований грамматик, задающих язык. Приведенные грамматики – определение, необходимые понятия.
26. Алгоритм преобразования грамматики к каноническому виду; алгоритмы устранения бесплодных и недостижимых символов, пустых правил и циклов.
27. Виды рекурсии в правилах грамматики. Алгоритм устранения левой рекурсии.
28. Виды распознавателей КС-языков – общая характеристика. Краткое описание возвратных методов анализа языка, их вычислительная сложность.
29. Нисходящий распознаватель КС-языков с возвратами. Алгоритм распознавателя с подбором альтернатив.
30. Восходящий распознаватель КС-языков с возвратами на основе алгоритма «сдвиг-свертка».
31. Распознаватели КС-языков без возвратов – основная идея, виды распознавателей.
32. Алгоритм нисходящего анализа текстов, работающий без возвратов. Метод рекурсивного спуска.
33. Класс LL-грамматик; определение, свойства, ограничения.
34. Класс LR-грамматик; определение, свойства, ограничения.
35. Грамматики предшествования как основа восходящего распознавателя языков и соответствующий алгоритм восходящего анализа текстов, работающий без возвратов.
36. Внутреннее представление программы, его формы.
37. Семантический анализ текста программы – сущность, назначение, этапы.
38. Синтаксически-управляемая трансляция – сущность, схема, особенности.
39. Оптимизация кода – сущность, критерии, примеры.

3.8 Перечень типовых простых практических заданий к экзамену

(для оценки умений)

Задание 1.

1. Выберите язык программирования по собственному выбору.
2. Напишите простую программу на выбранном языке программирования.
3. Объясните структуру программы и ее основную логику.
4. Измените программу так, чтобы она выполняла новую функцию.

Задание 2.

1. Выберите два различных языка программирования.
2. Сравните эти два языка программирования на основе критериев эффективности.
3. Исследуйте различные области применения этих языков.
4. Выпишите основные преимущества и недостатки каждого из языков.

Задание 3.

1. Исследуйте принципы работы Agile-методологии.
2. Разработайте простой план работы с использованием Agile-методологии на примере создания небольшого программного продукта.
3. Опишите, как процесс разработки будет адаптироваться к изменениям в требованиях с помощью Agile.
4. Рассмотрите возможные проблемы, которые могут возникнуть при использовании Agile, и предложите возможные решения.

Задание 4.

1. Исследуйте понятие DevOps и его ключевые принципы.
2. Напишите простую стратегию DevOps для небольшой команды разработчиков.
3. Опишите, как DevOps может улучшить процесс разработки и управления программным обеспечением.
4. Объясните, как инструменты управления версиями влияют на процесс разработки и поддержки программного обеспечения.

Задание 5.

1. Выберите язык программирования с поддержкой объектно-ориентированного программирования.
2. Разработайте простой класс с методами и свойствами в выбранном языке программирования.
3. Объясните, как использование классов и объектов повлияло на структуру и поведение программы.
4. Измените класс, добавив принципы наследования, полиморфизма и инкапсуляции.

Задание 6.

1. Изучите концепцию абстракции данных в объектно-ориентированном программировании.
2. Разработайте класс, использующий абстракцию данных, на языке программирования по вашему выбору.
3. Объясните, как использование абстракции данных помогло упростить и улучшить структуру программы.
4. Обсудите, как концепции объектно-ориентированного программирования могут быть применены в реальном проекте разработки программного обеспечения.

3.9 Перечень типовых практических заданий к экзамену

(для оценки навыков и (или) опыта деятельности)

Задание 1.

1. Исследуйте историю одного из языков программирования (на ваш выбор).
2. Проанализируйте основные парадигмы этого языка.

3. Опишите сферы применения выбранного языка.

Задание 2.

1. Изучите концепцию высокоуровневых и низкоуровневых языков программирования.
2. Выберите по одному языку из каждой категории и сравните их, учитывая синтаксис, доступные функции и области применения.

Задание 3.

1. Исследуйте и опишите один из современных подходов к разработке программного обеспечения (например, Agile или DevOps).
2. Приведите примеры применения этого подхода в реальных проектах.
3. Обсудите преимущества и недостатки выбранного подхода.

Задание 4.

1. Исследуйте вопросы безопасности в разработке программного обеспечения.
2. Опишите основные угрозы безопасности, с которыми сталкиваются разработчики.
3. Выберите несколько стратегий или инструментов для обеспечения безопасности программного обеспечения и проанализируйте их.

Задание 5.

1. Изучите принципы объектно-ориентированного программирования.
2. Опишите каждый из принципов с использованием примеров.
3. Обсудите важность каждого из этих принципов в контексте разработки программного обеспечения.

Задание 6.

1. Выберите объектно-ориентированный язык программирования и исследуйте его особенности и возможности.
2. Проанализируйте, как этот язык использует принципы объектно-ориентированного программирования.
3. Опишите типовые задачи, которые могут быть эффективно решены с помощью этого языка.

Задание 7.

1. Исследуйте концепцию классов и объектов в объектно-ориентированном программировании.
2. Проанализируйте отношения между классами и объектами и их роль в структуре программ.
3. Приведите примеры использования классов и объектов в программировании.

4. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности

В таблице приведены описания процедур проведения контрольно-оценочных мероприятий и процедур оценивания результатов обучения с помощью оценочных средств в соответствии с рабочей программой дисциплины.

Наименование оценочного средства	Описания процедуры проведения контрольно-оценочного мероприятия и процедуры оценивания результатов обучения
Собеседование	Собеседование, предусмотренное рабочей программой дисциплины, проводится на практическом занятии. Преподаватель на практическом занятии, предшествующем занятию проведения контроля, доводит до обучающихся тему, вопросы для подготовки к собеседованию. Результаты собеседования преподаватель доводит до обучающихся сразу после завершения собеседования
Доклад	Защита докладов, предусмотренных рабочей программой дисциплины, проводится во время практических занятий. Преподаватель на практическом занятии, предшествующем занятию проведения контроля, доводит до обучающихся: тему докладов и требования, предъявляемые к их выполнению и защите
Конспект	Защита конспектов, предусмотренных рабочей программой дисциплины, проводится во время практических занятий. Преподаватель на практическом занятии, предшествующем занятию проведения контроля, доводит до обучающихся: тему конспектов и требования, предъявляемые к их выполнению и защите
Лабораторная работа	Защита лабораторных работ проводится во время лабораторных занятий. Во время проведения защиты лабораторной работы пользоваться учебниками, справочниками, конспектами лекций, тетрадями не разрешено. Преподаватель на лабораторной работе, предшествующей занятию проведения защиты лабораторной работы, доводит до обучающихся: номер защищаемой лабораторной работы, время на защиту лабораторной работы. Преподаватель информирует обучающихся о результатах защиты лабораторной работы сразу после ее контрольно-оценочного мероприятия
Курсовая работа	Ход выполнения разделов курсовой работы в рамках текущего контроля оценивается преподавателем исходя из объемов выполненных работ в соответствие со шкалами оценивания. Преподаватель информирует обучающихся о результатах оценивания выполнения курсового проекта сразу после контрольно-оценочного мероприятия. В ходе защиты курсовой работы обучающийся делает доклад протяженностью 5 – 7 минут. Преподаватель ставит окончательную оценку за курсовую работу после завершения защиты, учитывая уровень ее защиты

Для организации и проведения промежуточной аттестации составляются типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы.

Перечень теоретических вопросов и типовые практические задания разного уровня сложности для проведения промежуточной аттестации обучающиеся получают в начале семестра через электронную информационно-образовательную среду ИрГУПС (личный кабинет обучающегося).

Описание процедур проведения промежуточной аттестации в форме экзамена и оценивания результатов обучения

Промежуточная аттестация в форме экзамена проводится путем устного собеседования по билетам или в форме компьютерного тестирования.

При проведении промежуточной аттестации в форме собеседования билеты составляются таким образом, чтобы каждый из них включал в себя теоретические вопросы и практические задания.

Билет содержит: два теоретических вопроса для оценки знаний. Теоретические вопросы выбираются из перечня вопросов к экзамену; два практических задания: одно из них для оценки умений (выбирается из перечня типовых простых практических заданий к экзамену); другое практическое задание для оценки навыков и (или) опыта деятельности (выбираются из перечня типовых практических заданий к экзамену).

Распределение теоретических вопросов и практических заданий по экзаменационным билетам находится в закрытом для обучающихся доступе. Разработанный комплект билетов (25-30 билетов) не выставляется в электронную информационно-образовательную среду ИрГУПС, а хранится на кафедре-разработчике фондов оценочных средств.

На экзамене обучающийся берет билет, для подготовки ответа на экзаменационный билет обучающемуся отводится время в пределах 45 минут. В процессе ответа обучающегося на вопросы и задания билета, преподаватель может задавать дополнительные вопросы.

Каждый вопрос/задание билета оценивается по четырехбалльной системе, а далее вычисляется среднее арифметическое оценок, полученных за каждый вопрос/задание. Среднее арифметическое оценок округляется до целого по правилам округления

При проведении промежуточной аттестации в форме компьютерного тестирования вариант тестового задания формируется из фонда тестовых заданий по дисциплине случайным образом, но с условием: 50 % заданий должны быть заданиями открытого типа и 50 % заданий – закрытого типа.

Образец экзаменационного билета

ИРГУПС 20__-20__ учебный год	**Экзаменационный билет № 1** **по дисциплине «Теория языков программирования и методы трансляции»**	Утверждаю: Заведующий кафедрой «_____» ИрГУПС _____
1. Формальные языки и грамматики. Классификация грамматик и языков. 2. Схема работы компилятора 3. Какой язык генерируется следующими грамматиками? a) S→0S1 | 01 b) S→+SS | -SS | a c) S→S(S)S | ε d) S→aSbS | aSbS | ε e) S→a | S+S | SS | S* | (S)		