

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Алтайский государственный технический университет им. И.И. Ползунова»

СОГЛАСОВАНО

Декан ФИТ

А.С. Авдеев

Рабочая программа дисциплины

Код и наименование дисциплины: **Б1.В.6 «Конструирование компиляторов»**

Код и наименование направления подготовки (специальности): **09.04.04**

Программная инженерия

Направленность (профиль, специализация): **Разработка программно-информационных систем**

Статус дисциплины: **часть, формируемая участниками образовательных отношений**

Форма обучения: **очная**

Статус	Должность	И.О. Фамилия
Разработал	профессор	Е.Н. Крючкова
Согласовал	Зав. кафедрой «ПМ»	Е.Г. Боровцов
	руководитель направленности (профиля) программы	С.М. Старолетов

г. Барнаул

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Компетенция	Содержание компетенции	Индикатор	Содержание индикатора
ПК-6	Способен проектировать трансляторы и интерпретаторы языков программирования	ПК-6.1	Анализирует и выбирает методы проектирования трансляторов и интерпретаторов языков программирования
		ПК-6.2	Проектирует трансляторы и интерпретаторы языков программирования

2. Место дисциплины в структуре образовательной программы

Дисциплины (практики), предшествующие изучению дисциплины, результаты освоения которых необходимы для освоения данной дисциплины.	
Дисциплины (практики), для которых результаты освоения данной дисциплины будут необходимы, как входные знания, умения и владения для их изучения.	Выпускная квалификационная работа, Преддипломная практика

3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося

Общий объем дисциплины в з.е. /час: 5 / 180

Форма промежуточной аттестации: Экзамен

Форма обучения	Виды занятий, их трудоемкость (час.)				Объем контактной работы обучающегося с преподавателем (час)
	Лекции	Лабораторные работы	Практические занятия	Самостоятельная работа	
очная	16	32	0	132	62

4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

Форма обучения: очная

Семестр: 2

Лекционные занятия (16ч.)

1. Структура компилятора {лекция с разбором конкретных ситуаций} (2ч.)[2,4,6,7] Понятие компиляции и интерпретации. Методы проектирования трансляторов и интерпретаторов языков программирования. Лексический, синтаксический и семантический анализ.

КС-грамматики языков программирования.

Контекстные условия.

Реализация лексического уровня.

2. Синтаксический анализ {лекция с разбором конкретных ситуаций} (4ч.)[2,3] Нисходящий и восходящий синтаксический анализ.

Рекурсивные и магазинные методы синтаксического анализа.

Метод рекурсивного спуска.

LL(k)-анализаторы: определение, алгоритм построения управляющей таблицы, конфликты и их устранение.

Программирование LL(k)-анализатора.

S-анализаторы.

Анализаторы предшествования: определение, алгоритм построения управляющей таблицы,

конфликты и их устранение.

Программирование анализатора предшествования.

LR(k)-анализаторы: определение, алгоритм построения управляющей таблицы. Программирование LR(k)-анализатора.

3. Синтаксически управляемый перевод {лекция с разбором конкретных ситуаций} (4ч.)[2,3] Определение синтаксически управляемого перевода.

Операционные символы и семантические подпрограммы.

Понятие промежуточного кода. Способы представления дерева разбора: древовидная структура, ПОЛИЗ, тетрады и триады.

Примеры представления в промежуточном коде конструкций языка программирования.

4. Оптимизация кода {лекция с разбором конкретных ситуаций} (2ч.)[2,3] Граф управления программой (Flow Diagram).

Выделение линейных участков, циклов, ветвлений.

Оптимизация выражений: вычисление, совпадающие фрагменты, Агрегаты аддитивных и мультипликативных операций.

Оптимизация ветвлений и циклов.

Оптимизация функций и их вызовов. Оптимизация хвостовой рекурсии.

Удаление недостижимого кода.

5. Генерация кода {лекция с разбором конкретных ситуаций} (2ч.)[2]

Структура генерируемого кода,

назначение и структура сегментов ассемблерного кода.

Назначение регистров.

Распределение памяти, выделение памяти для локальных и глобальных данных.

Пролог и эпилог функции.

Генерация уникальных имен.

Команды с вещественными и целыми даннами.

Ассемблерный код для функций и их вызовов. Передача параметров через регистры и стек.

6. Автоматизация проектирования трансляторов {лекция с разбором конкретных ситуаций} (2ч.)[2,5] Метода автоматизации построения трансляторов.

Ранние генераторы Lex и Yacc.

Система ANTLR: язык описания грамматики, использование средств отладки, генерация кода, работа с семантическими функциями, методы listener и visitor.

Лабораторные работы (32ч.)

1. Нисходящий анализатор {метод кейсов} (6ч.)[2,4,6] Описать лексический уровень ЯП и реализовать лексический анализатор.

Описать синтаксический уровень ЯП в форме КС-грамматики.

Построить управляющую таблицу LL(1)-анализатора.

Решить проблему разрешения конфликтов при их наличии.

Реализовать LL(1)-анализатор.

2. Восходящий анализ {метод кейсов} (4ч.)[1,2,3,4] Описать лексический уровень ЯП и реализовать лексический анализатор.

Описать синтаксический уровень ЯП в форме КС-грамматики.

Построить управляющую таблицу LL(1)-анализатора.

Решить проблему разрешения конфликтов при их наличии.

Реализовать LL(1)-анализатор.

3. Синтаксически управляемый перевод {метод кейсов} (6ч.)[1,2] Предложить список семантических подпрограмм,

предназначенных для проверки контекстных условий

Для нисходящей стратегии разбора предложить СУ-перевод проверки контекстных условий

Предложить СУ-перевод проверки контекстных условий для восходящей стратегии разбора

Реализовать класс семантического дерева

4. Генератор промежуточного кода {метод кейсов} (6ч.)[1,2,3] Предложить конструкцию промежуточного кода.

Дополнить СУ-перевод функциями генерации промежуточного кода.

Встроить семантические функции перевода в программу нисходящего анализатора.

Проверить работоспособность на правильных и ошибочных тестах (дублирование идентификатора, неописанные данные, неверное использование данных).

5. Оптимизация кода. {метод кейсов} (4ч.)[1,2,3] Рассмотреть примеры

оптимизации выражений на реальных компиляторах (например, O2 трансляторов gcc или cl).

Предложить 2 - 3 метода оптимизации выражений

Реализовать предложенную оптимизацию.

Оценить эффективность предложенных алгоритмов по сравнению с известными компиляторами.

Оптимизация сложных операторов.

Рассмотреть примеры оптимизации конструкций на реальных компиляторах (например, /o2 VS) - вызовы функций, рекурсия, циклы, лишние вычисления и т.п.

Предложить метод оптимизации наиболее сложной конструкции в транслируемом ЯП.

Реализовать предложенную оптимизацию.

Оценить эффективность предложенных алгоритмов по сравнению с известными компиляторами.

6. Генерация ассемблерного кода выражений {метод кейсов} (6ч.)[1,2,3]

Адресация данных (локальные и глобальные данные).

Рассмотреть способы адресации данных на реальных компиляторах (например, gcc или cl).

Выбрать способы адресации данных для реализуемого Вами компилятора.

В семантическое дерево добавить поле для хранения представления адреса.

Заполнить адресные поля для всех объектов семантического дерева.

Предложить форму ассемблерных команд для триад.

разных типов, в том числе для триад с непосредственными операндами, операндами-константами, операндами-ссылками.

Выбрать для реализации 3-4 кода операций в триадах.

Выбрать способ выделения регистров для промежуточных вычислений. Реализовать генерацию ассемблерного кода для выражений.

Курсовые работы (72ч.)

1. Выполнение курсового проекта в соответствии с заданием(72ч.)[2,3,5,6]

Самостоятельная работа (132ч.)

1. Подготовка к лекциям, проработка лекционного материала(16ч.)[2]

2. Подготовка к лабораторным работам(80ч.)[1,2,3]

3. Подготовка к экзамену(36ч.)[1,2,3,4,5,6,7]

5. Перечень учебно-методического обеспечения самостоятельной работы обучающихся по дисциплине

Для каждого обучающегося обеспечен индивидуальный неограниченный доступ к электронно-библиотечным системам: Лань, Университетская библиотека он-лайн, электронной библиотеке АлтГТУ и к электронной информационно-образовательной среде:

1. Крючкова Е. Н.

Методические указания к выполнению лабораторных работ по курсу "Конструирование компиляторов" для студентов направления 09.04.04 "Программная инженерия" (магистратура). - Барнаул, 2020 - 11с.

Прямая

ссылка:

http://elib.altstu.ru/eum/download/pm/Kruchkova_KonstrKomp_lr_mu.pdf

6. Перечень учебной литературы

6.1. Основная литература

2. Крючкова Е. Н. Основы теории конструирования компиляторов: Учебно-методическое пособие.- Барнаул: АлтГТУ, 2020. - 405с. Прямая ссылка: http://elib.altstu.ru/eum/download/pm/Kruchkova_OTKK_up.pdf

6.2. Дополнительная литература

3. Вирт, Н. Построение компиляторов [Электронный ресурс] / Н. Вирт. - Электрон. дан. - Москва : ДМК Пресс, 2010. - 192 с. - Режим доступа: <https://e.lanbook.com/book/1262>. - Загл. с экрана.

4. Авдошин, С.М. Дискретная математика. Формально-логические системы и языки [Электронный ресурс] / С.М. Авдошин, А.А. Набебин. - Электрон. дан. - Москва : ДМК Пресс, 2018. - 390 с. - Режим доступа: <https://e.lanbook.com/book/100912>. - Загл. с экрана.

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

5. ANTLR [Электронный ресурс] // – Режим доступа: <https://www.antlr.org/>, свободный.

6. Математическая теория формальных языков: Материалы сайта "Интернет университет информационных технологий" - сетевой адрес: <http://www.intuit.ru/department/algorithms/mathformlang/>

7. Программы дистанционного обучения в НОУ «ИНТУИТ»: Автоматное программирование: анализ задачи - сетевой адрес: <http://www.intuit.ru/department/se/progstyles/9/>

8. Фонд оценочных материалов для проведения текущего контроля успеваемости и промежуточной аттестации

Содержание промежуточной аттестации раскрывается в комплекте контролирующих материалов, предназначенных для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС, которые хранятся на кафедре-разработчике РПД в печатном виде и в ЭИОС.

Фонд оценочных материалов (ФОМ) по дисциплине представлен в приложении А.

9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

Для успешного освоения дисциплины используются ресурсы электронной информационно-образовательной среды, образовательные интернет-порталы, глобальная компьютерная сеть Интернет. В процессе изучения дисциплины происходит интерактивное взаимодействие обучающегося с преподавателем через личный кабинет студента.

№пп	Используемое программное обеспечение
1	Acrobat Reader
2	Eclipse IDE
3	Java Runtime Environment
4	LibreOffice
5	MASM32
6	Microsoft Office
7	Mozilla Firefox
8	Python
9	Qt Creator Open Source
10	Visual Studio
11	Windows
12	Антивирус Kaspersky

№пп	Используемые профессиональные базы данных и информационные справочные системы
1	IEEE Xplore - Интернет библиотека с доступом к реферативным и полнотекстовым статьям и материалам конференций. Бессрочно без подписки (https://ieeexplore.ieee.org/Xplore/home.jsp)
2	Springer - Издательство с доступом к реферативным и полнотекстовым материалам журналов и книг (https://www.springer.com/gp https://link.springer.com/)
3	Ассоциация Разработчиков Программных Продуктов «Отечественный софт» Общедоступная база данных профессиональных сообществ и их членов (https://www.arppsoft.ru/)
4	Бесплатная электронная библиотека онлайн "Единое окно к образовательным ресурсам" для студентов и преподавателей; каталог ссылок на образовательные интернет-ресурсы (http://Window.edu.ru)
5	На сайте проекта OpenNet размещается информация о Unix системах и открытых технологиях для администраторов, программистов и пользователей (http://www.opennet.ru/)
6	Национальная электронная библиотека (НЭБ) — свободный доступ читателей к

№пп	Используемые профессиональные базы данных и информационные справочные системы
	фондам российских библиотек. Содержит коллекции оцифрованных документов (как открытого доступа, так и ограниченных авторским правом), а также каталог изданий, хранящихся в библиотеках России. (http://нэб.рф/)
7	Программа Microsoft и интернет-ресурс, содержащий техническую информацию, новости и предстоящие события для профессионалов в сфере информационных технологий. На данный момент представляет собой сборник технической информации на русском языке для IT-специалистов (https://technet.microsoft.com/ru-ru/ https://docs.microsoft.com/ru-ru/welcome-to-docs)

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

Наименование специальных помещений и помещений для самостоятельной работы
учебные аудитории для проведения учебных занятий
помещения для самостоятельной работы

Материально-техническое обеспечение и организация образовательного процесса по дисциплине для инвалидов и лиц с ограниченными возможностями здоровья осуществляется в соответствии с «Положением об обучении инвалидов и лиц с ограниченными возможностями здоровья».