



Министерство образования Российской Федерации
Московский Государственный Технический Университет им. Н.Э. Баумана
Факультет «Информатика и системы управления»
Кафедра «Теоретическая информатика и компьютерные технологии»

УТВЕРЖДАЮ
Заведующий кафедрой ИУ-9
_____ (И.П. Иванов)
«__» _____ 20__ г.

Техническое задание
на дипломную работу
«Разработка оптимизирующего компилятора с открытой архитектурой для
императивного языка программирования»

Исполнитель: А. М. Бобух
Группа: ИУ9-111

СОГЛАСОВАНО
Руководитель
квалификационной работы
_____ (С.Ю. Скоробогатов)
«__» _____ 20__ г.

1. Введение.

- 1.1. Наименование работы «Разработка оптимизирующего компилятора с открытой архитектурой для императивного языка программирования».
- 1.2. Исполнитель дипломной работы — А. М. Бобух.

2. Основания для разработки.

- 2.1. Основанием для разработки дипломной работы является решение кафедры ИУ-9 от «___» _____ 20___ г., протокол № ____.

3. Используемые понятия.

Понятия, используемые в техническом задании, приведены в таблице Таблица 1.

Таблица 1 Понятия, используемые в техническом задании.

Понятие	Определение
Промежуточное представление (IR = Intermediate Representation)	Представление программы, удобное для проведения оптимизаций
Машинно-зависимое промежуточное представление (MSIR = Machine Specific Intermediate Representation)	Модифицированное промежуточное представление, зависящее от конкретной платформы, для которой производится компиляция программы.
Граф потока управления (CFG = Control Flow Graph)	Множество всех возможных путей передачи управления между базовыми блоками, представленных в виде графа.
Локальная оптимизация	Оптимизация промежуточного представления программы, при которой происходит сопоставления блоков кода с шаблонами и последующей заменой этих блоков упрощенной реализацией.
Constant folding-оптимизация	Оптимизация константных выражений на этапе компиляции программы.

4. Цель и назначения разработки.

- 4.1. Назначение: Разрабатываемый компилятор должен компилировать программы из учебного императивного языка программирования в язык ассемблера.
- 4.2. Цель: Получить на кафедре ИУ-9 работающий макет компилятора для проведения исследований и учебных занятий.

5. Требования к программному комплексу.

5.1. Составные части компилятора:

5.1.1. Front end:

- лексический анализатор программ на разрабатываемом языке [1];
- синтаксический анализатор программ на разрабатываемом языке;
- семантический анализатор программ на разрабатываемом языке;
- генератор промежуточного представления (IR) в виде трехадресного кода.

5.1.2. Middle end:

- библиотека для работы с графами;
- построение CFG [2];
- оптимизатор.

5.1.3. Back end:

- транслятор из IR в MSIR;
- компилятор из MSIR в язык ассемблера для платформы AMD64.

5.2. Требования к функциональным характеристикам.

5.2.1. Лексического анализатора:

- разбиение исходного кода программы на языке программирования на лексемы.

5.2.2. Синтаксического анализатора:

- построение дерева синтаксического разбора программы;

5.2.3. Семантического анализатора:

- многопроходный семантический анализ;
- построение множества таблиц символов программы;

5.2.4. Генератора промежуточного представления:

- построение промежуточного представления в виде трехадресного кода по дереву синтаксического разбора и таблицам символов программы;

5.2.5. Библиотека для работы с графами:

- поддержка разреженных графов;
- поддержка плотных графов;
- поддержка итераторов, свойств вершин, свойств ребер графа;
- алгоритмы обхода графа в глубину/ширину;
- поиск компонент сильной связности в графах;
- алгоритм поиска критических путей в графе;
- построение деревьев доминаторов и постдоминаторов;
- визуализация графов.

5.2.6. Построение графа потока управления:

- реализация графа потока управления на основании библиотеки работы с графами;
- построение графа потока управления.

5.2.7. Оптимизатор:

- реализация менеджера фаз;
- элементарные оптимизации:
 - локальная оптимизация;
 - constant folding-оптимизация.

5.2.8. Транслятор из IR в MSIR:

- машинная модель;

- спецификация модели окружения операционной системы (OSEM);
 - спецификация машинно-зависимого промежуточного представления (MSIR).
- 5.2.9. Требование к аппаратному и программному обеспечению.
- Требование к аппаратному обеспечению:
 - Компьютер на базе платформы AMD64.
- 5.2.10. Требования к программному обеспечению:
- Операционная система Linux на основе ядра с версией 2.6;
 - Установленная Java Runtime Environment (JRE).
- 5.2.11. Надёжность.
- Модули компилятора должны правильно выполнять все преобразования исходного кода, не нарушать его работоспособности и не изменять его семантики.
 - Работоспособность модулей компилятора должна быть проверена путём написания тестовых программ на исходном языке программирования и их компиляции с включенными и выключенными оптимизациями.
- 5.2.12. Условия эксплуатации.
- Условия эксплуатации должны соответствовать требованиям по условиям эксплуатации технических средств, выполняющих программу.
- 5.3. Требование к программной документации.
- 5.3.1. Программная документация должна быть выполнена в соответствии с требованиями ЕСПД.
- 5.3.2. На программный пакет должен быть разработан следующий комплект документации:
- Техническое задание (ТЗ);
 - Пояснительная записка, включающая руководство пользователя;
 - UML-диаграммы типов данных.
- 5.4. Требования по эргономике и эстетике не предъявляются.
- 5.5. Требования по маркировке, упаковке и хранению.
- 5.5.1. Требования по маркировке, упаковке и хранению должны соответствовать требованиям по маркировке, упаковке и хранению носителей информации, используемых для хранения программы.
- 5.6. Требования к транспортировке и хранению.
- 5.6.1. Требования к транспортировке и хранению должны соответствовать требованиям к транспортировке и хранению носителей информации, используемых для хранения программы.
- 5.7. Специальные требования.
- 5.7.1. Не должна производиться компиляция программ, содержащих ошибки.
- 6. Технико-экономические требования.**
- 6.1. Проводится расчет технико-экономической эффективности разработки.
- 7. Стадии и этапы разработки.**
- 7.1. Стадии и этапы разработки приведены в таблице Таблица 2.

Таблица 2 Стадии и этапы разработки дипломной работы.

Стадии работ	Этапы работ и их содержание	Исполнитель	Сроки исполнения		Результаты работ
			Начало	Конец	
Подготовительная	Разработка ТЗ.	Бобух	02.09.11	21.09.11	ТЗ утверждено.
	Сбор информации о существующих разработках.		21.09.11	01.10.11	Обзор существующих разработок.
Основная	Реализация front end.	Бобух	25.09.11	27. 10.11	Front end компилятора.
	Реализация middle end.		01.11.12	15.01.12	Middle end компилятора.
	Реализация back end.		15.01.12	15.03.12	Back end компилятора.
	Разработка программной документации.		15.03.12	23.04.12	Руководство пользователя.
	Проведение технико-экономических расчетов.		01.04.12	10.05.12	Экономическая часть работы.
Заключительная	Оформление пояснительной записки.	Бобух	10.05.12	31.05.12	Пояснительная записка.
	Разработка презентации.		17.05.12	31.05.12	Презентация.

8. Порядок контроля и приемки.

1. Приемка работ осуществляется в два этапа:
 - 8.1.1. предварительная защита перед комиссией представителей кафедры ИУ-9 до 1 июня 2012 г.
 - 8.1.2. основная защита на Государственной Аттестационной Комиссии в период с 10 по 25 июня 2012 г.

9. Список литературы.

1. Скоробогатов С. Ю. Лекции по курсу «Введение в конструирование компиляторов». М., 2010.
2. Ахо, А. В., Лам, М. С., Сети, Р., Ульман, Д. Д. Компиляторы: принципы, технологии и инструментари, 2-е изд. Пер. с англ. / А. В. Ахо, М. С. Лам, Р. Сети, Д. Д. Ульман. – М.: Вильямс, 2008.

10. В ходе работы отдельные пункты технического задания могут уточняться, дополняться, исключаться.

Консультант по конструкторской части
_____ (С.Ю. Скоробогатов)
«__» _____ 2011 г.

Студент МГТУ имени Н.Э. Баумана
_____ (А.М. Бобух)
«__» _____ 2011 г.