

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Департамент программной инженерии

Согласовано

Профессор департамента
программной инженерии
факультета компьютерных наук
канд. техн. наук

_____ Гринкруг Е. М.
" " _____ 2017 г

Утверждаю

Академический руководитель
образовательной программы
«Программная инженерия»
профессор департамента программной
инженерии канд. техн. наук

_____ Шилов В. В.
" " _____ 2017 г

**АЛГОРИТМ ДЛЯ ГЛОБАЛЬНОГО РАСПРЕДЕЛЕНИЯ РЕГИСТРОВ В
ЭМУЛЯТОРЕ QEMU И ЕГО РЕАЛИЗАЦИЯ**

Техническое задание

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.509000 ТЗ 01-1

Студент группы БПИ 151 НИУ ВШЭ
_____ Абрамов А.М.
" " _____ 2017 г

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл.	

2017

УТВЕРЖДЕНО
RU.17701729.509000 ТЗ 01-1

**АЛГОРИТМ ДЛЯ ГЛОБАЛЬНОГО РАСПРЕДЕЛЕНИЯ РЕГИСТРОВ В
ЭМУЛЯТОРЕ QEMU И ЕГО РЕАЛИЗАЦИЯ**

Техническое задание

RU.17701729.509000 ТЗ 01-1

Листов 16

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Содержание

1 Введение	3
1.1 Наименование	3
1.2 Краткая характеристика	3
2 Основания для разработки	4
2.1 Документ, на основании которого ведется разработка	4
2.2 Наименование темы разработки	4
3 Назначение разработки	5
3.1 Функциональное назначение	5
3.2 Эксплуатационное назначение	5
4 Требования к программному изделию	6
4.1 Требования к функциональным характеристикам	6
4.1.1 Состав выполняемых функций	6
4.1.2 Организация входных и выходных данных	6
4.1.3 Прочие требования	6
4.2 Требования к временным характеристикам	6
4.3 Требования к интерфейсу	6
4.4 Требования к надежности	7
4.4.1 Обеспечение устойчивого функционирования программы	7
4.4.2 Время восстановления после отказа	7
4.4.3 Отказы из-за некорректных действий оператора	7
4.5 Требования к условиям эксплуатации	7
4.5.1 Вид обслуживания	7
4.5.2 Численность и квалификация персонала	7
4.6 Требования к составу и параметрам технических средств	7
4.7 Требования к информационной и программной совместимости	8
4.8 Требования к упаковке	8
5 Требования к программной документации	9
5.1 Предварительный состав программной документации	9

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

6	Технико-экономические показатели	10
6.1	Орентирующая экономическая эффективность	10
6.2	Экономические преимущества разработки	10
7	Стадии и этапы разработки	11
7.1	Необходимые стадии разработки	11
7.1.1	Стадия разработки технического задания:	11
7.1.2	Стадия разработки технического проекта:	11
7.1.3	Стадия разработки рабочего проекта:	11
7.2	Сроки работ и исполнители	12
8	Порядок контроля и приемки	13
8.1	Виды испытаний	13
8.2	Требования к приемке работы	13
9	Приложение 1. Терминология	14
9.1	Терминология	14
10	Приложение 2. Список используемой литературы	15
10.1	Список используемой литературы	15

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

1. Введение

1.1. Наименование

Наименование: «Алгоритм для глобального распределения регистров в эмуляторе QEMU и его реализация».

Наименование на английском: «Algorithm for global management of registers in the QEMU emulator and its implementation».

1.2. Краткая характеристика

Цель работы - составить и реализовать наиболее оптимальный алгоритм для глобального распределения регистров в эмуляторе QEMU. В задачи работы входит рассмотрение уже существующих алгоритмов, разработка наиболее оптимального алгоритма и его реализация. Рассмотрение уже существующих алгоритмов для глобального распределения регистров выявляет их общие характеристики, а также их слабые и сильные стороны. Основываясь на данном анализе разрабатывается наиболее оптимальный алгоритм. Реализация алгоритма в эмуляторе QEMU позволит пользователям эмулятора эффективнее использовать ресурсы системы. В состав работы также входит создание демонстрационных исходных данных (файлов) для измерения эффективности работы алгоритма.

Демонстрационные исходные данные представляют собой исполняемую программу, удовлетворяющую требованиям входных данных, которая запускается в эмуляторе QEMU.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

2. Основания для разработки

2.1. Документ, на основании которого ведется разработка

Разработка программы ведется на основании приказа №6.18.1-02/1112-19 от 11.12.2015 «Об утверждении тем, руководителей курсовых работ студентов образовательной программы Программная инженерия факультета компьютерных наук» в соответствии с учебным планом подготовки бакалавров по направлению «Программная инженерия», факультета Компьютерных наук, Национального исследовательского университета «Высшая школа экономики»

2.2. Наименование темы разработки

Наименование: «Алгоритм для глобального распределения регистров в эмуляторе QEMU и его реализация».

Наименование на английском: «Algorithm for global management of registers in the QEMU emulator and its implementation».

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

3. Назначение разработки

3.1. Функциональное назначение

Функциональным назначением разработки является предоставление пользователю возможности более быстро и эффективно эмулировать системы используя эмулятор QEMU, а также отчета по анализу уже существующих алгоритмов, их сильных и слабых сторон.

3.2. Эксплуатационное назначение

Реализованный алгоритм предназначен для включения в сборку программы QEMU на операционной системе Linux. Алгоритм может использоваться любым пользователем желающим ускорить работу эмулятора QEMU и снизить требования эмулятора к системным ресурсам. Исходный код может использоваться в учебных целях как пример реализации алгоритма тесно взаимодействующего с внутренними механизмами QEMU.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

4. Требования к программному изделию

4.1. Требования к функциональным характеристикам

4.1.1. Состав выполняемых функций

1. Интеграция в QEMU версии 2.10 или позднее.
2. Предоставление более оптимального алгоритма чем уже существующие решения.

4.1.2. Организация входных и выходных данных

Входными данными для работы алгоритма является абстрактное синтаксическое дерево базовых блоков программы в формате внутреннего представления эмулятора QEMU. Также для работы алгоритма необходима исполняемая программа, которая может быть запущена в эмуляторе QEMU. Входной файл исполняемой программы может быть создан в любой среде разработки на платформе которую поддерживает эмулятор QEMU, например x86_64 с операционной системой Linux.

1. Из-за огромного количества подходов, анализировать их все не представляется возможным. Поэтому алгоритм должен быть оптимальнее только тех алгоритмов, которые рассмотрены в качестве уже существующих.
2. Файл программы должен представлять собой либо ядро системы скомпилированное под платформу которая поддерживается эмулятором QEMU, либо должен представлять собой исполняемый файл предназначенный для запуска в userspace операционной системы Linux на архитектуре x86_64.
3. Файл программы должен быть предоставлен в формате ELF.

Выходными данными для алгоритма является граф оптимального глобального распределения регистров эмулятора QEMU.

4.1.3. Прочие требования

1. Отсутствие влияния на корректное исполнение программы.
2. Наличие пояснений и обоснование оптимальности алгоритма по сравнению с уже существующими решениями.

4.2. Требования к временным характеристикам

1. Алгоритм должен работать быстрее чем уже существующие решения.

4.3. Требования к интерфейсу

Требования к интерфейсу не предъявляются.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

4.4. Требования к надежности

4.4.1. Обеспечение устойчивого функционирования программы

Алгоритм не должен вне зависимости от входных данных или действий оператора завершаться аварийно. При некорректно введенных параметрах должно отображаться сообщение об ошибке.

4.4.2. Время восстановления после отказа

Требования к восстановлению после отказа не предъявляются.

4.4.3. Отказы из-за некорректных действий оператора

Требования к отказу из-за некорректных действий оператора не предъявляются.

4.5. Требования к условиям эксплуатации

4.5.1. Вид обслуживания

Не требует каких-либо видов обслуживания.

4.5.2. Численность и квалификация персонала

Минимальное количество персонала, требуемого для работы: 1 оператор. Пользователь эмулятора QEMU должен иметь образование не ниже среднего, обладать практическими навыками работы с компьютером.

4.6. Требования к составу и параметрам технических средств

Для оптимальной работы алгоритма в эмуляторе QEMU необходимо учесть следующие системные требования:

1. Компьютер, оснащенный:
 - (а) 64-разрядный (x86_64) процессор с тактовой частотой 1 гигагерц (ГГц) или выше;
 - (б) 1 ГБ оперативной памяти (ОЗУ);
 - (с) 1.5 ГБ свободного места на жестком диске;
2. Монитор
3. Мышь
4. Клавиатура

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

4.7. Требования к информационной и программной совместимости

Реализация алгоритма для распределения глобальных регистров в эмуляторе QEMU обязательно должен быть написан с использованием языка С. Алгоритм должен включаться в сборку эмулятора QEMU и исправно работать на 64-разрядных процессорах под операционной системой Linux.

4.8. Требования к упаковке

Реализация алгоритма поставляется в виде патча для исходного кода эмулятора QEMU на внешнем носителе информации – USB флеш накопителе. На нем должны содержаться документация по алгоритму, исходный код для алгоритма, доклад сравнивающий характеристики работы алгоритма с другими уже существующими решениями.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

5. Требования к программной документации

5.1. Предварительный состав программной документации

В обязательном порядке должны входить:

1. Техническое задание (ГОСТ 19.201-78)
2. Пояснительная записка (ГОСТ 19.404-79)
3. Руководство оператора (ГОСТ 19.505-79)
4. Программа и методика испытаний (ГОСТ 19.301-79*)
5. Текст программы (ГОСТ 19.401-78*)

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

6. Техничко-экономические показатели

6.1. Оринтировочная экономическая эффективность

Оринтировочная экономическая эффективность не рассчитывается.

6.2. Экономические преимущества разработки

Данная разработка позволяет использовать современные математические исследования в области графов и компиляторных технологий для предоставления наиболее оптимального на данный момент алгоритма по распределению глобальных регистров в эмуляторе QEMU.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

7. Стадии и этапы разработки

7.1. Необходимые стадии разработки

7.1.1. Стадия разработки технического задания:

1. Этап обоснования необходимости разработки программы:
 - (а) постановка задачи.
 - (б) сбор исходных материалов.
2. Этап разработки и утверждения технического задания:
 - (а) определение требований к алгоритму.
 - (б) определение стадий, этапов и сроков разработки программы и документации на нее.
 - (с) согласование и утверждение технического задания.

7.1.2. Стадия разработки технического проекта:

1. Этап исследования уже существующих решений:
 - (а) поиск уже созданных решений
 - (б) изучение их структуры и архитектуры
 - (с) анализ их рабочих характеристик
2. Этап разработки технического проекта:
 - (а) разработка алгоритма
 - (б) разработка структуры и архитектуры частей алгоритма.
 - (с) анализ оптимальности найденного алгоритма
3. Этап утверждения технического проекта:
 - (а) разработка плана мероприятий по разработке программы
 - (б) разработка пояснительной записки.

7.1.3. Стадия разработки рабочего проекта:

1. Этап разработки программы:
 - (а) непосредственное программирование и отладка алгоритма.
2. Этап разработки программной документации:
 - (а) разработка следующих программных документов в соответствии с требованиями: техническое задание, пояснительная записка, руководство оператора, программа и методика испытания, текст программы, все в соответствии с требованиями ГОСТ 19.101-77.
3. Этап испытания программы:
 - (а) разработка, согласование и утверждение программы и методики испытаний.
 - (б) испытания программы.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

- (с) защита презентации, сдача разработанной документации.
- (d) корректировка программы и программной документации по результатам испытаний.

7.2. Сроки работ и исполнители

Алгоритм должен быть разработан к 10 мая 2018 года, студентом группы БПИ151 Абрамовым Артемом.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

8. Порядок контроля и приемки

8.1. Виды испытаний

Контроль и приемка разработки осуществляются в соответствии с разработанным исполнителем и согласованным с заказчиком документом «Алгоритм для глобального распределения регистров в эмуляторе QEMU и его реализация» Программа и методика испытаний по (ГОСТ 19.301-79*).

8.2. Требования к приемке работы

Акт приемки-сдачи программы между исполнителем и заказчиком в эксплуатацию происходит при полной работоспособности алгоритма, при выполнении указанных в настоящем документе функций и требований, при наличии документации к программе, выполненной в соответствии с требованиями настоящего технического задания.

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

9. Приложение 1. Терминология

9.1. Терминология

Архитектура набора команд (англ. instruction set architecture, ISA) часть архитектуры компьютера, определяющая программируемую часть ядра микропроцессора. На этом уровне определяются реализованные в микропроцессоре конкретного типа

контрольный таймер, англ. Watchdog timer аппаратно реализованная схема контроля над зависанием системы. Представляет собой таймер, который периодически сбрасывается контролируемой системой. Если сброса не произошло в течение некоторого интервала времени, происходит принудительная перезагрузка системы. В некоторых случаях сторожевой таймер может посылать системе сигнал на перезагрузку («мягкая» перезагрузка), в других же — перезагрузка происходит аппаратно (замыканием сигнального провода RST или подобного ему).

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

10. Приложение 2. Список используемой литературы

10.1. Список используемой литературы

1. Quality and Speed in Linear-scan Register Allocation
Omri Traub, Glenn Holloway, Michael D. Smith
Harvard University, Division of Engineering and Applied Sciences Cambridge, MA 02138
2. REGISTER ALLOCATION & SPILLING VIA GRAPH COLORING
G. J. Chaitin
IBM Research, P.O.Box 218, Yorktown Heights, NY 10598
3. Linear Scan Register Allocation
MASSIMILIANO POLETTO, Laboratory for Computer Science, MIT
VIVEK SARKAR IBM Thomas J. Watson Research Center

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 T3 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата

Лист регистрации изменений

Изм.	Номера листов (страниц)				Всего листов (страниц) в докум.	№ докум.	Входящий № сопроводительного докум. и дата	Подпись	Дата
	измененных	замененных	новых	аннулированных					

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.509000 ТЗ 01-1				
Инв. №подл.	Подп. и дата	Взам. инв. №	Инв. №дубл.	Подп. и дата