

«

»

“ ”

“ ”

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ Архитектура вычислительных систем и компьютерных сетей

: 02.03.03

, :

: 1, : 2

		2
1	()	2
2		72
3	, .	42
4	, .	36
5	, .	0
6	, .	0
7	, .	6
8	, .	2
9	, .	4
10	, .	30
11	(, ,)	
12		

(): 02.03.03

222 12.03.2015 ., : 07.04.2015 .

: 1,

(): 02.03.03

, 5 20.06.2017
, 4 20.06.2017

, 6 21.06.2017

:

,

:

,

:

. . .

1.

1.1

Компетенция ФГОС: ОПК.5 владение информацией о направлениях развития компьютеров с традиционной (нетрадиционной) архитектурой; о тенденциях развития функций и архитектур проблемно-ориентированных программных систем и комплексов; в части следующих результатов обучения:	
3.	,
1.	
2.	
Компетенция НГТУ: ПК.9.В/ОУ способность к выбору архитектуры и комплексирования современных компьютеров, систем, комплексов и сетей; в части следующих результатов обучения:	
3.	

2.

2.1

	(
	,	
	,	
)	

.9. / . 3	
1.О моделях вычислений.	;
2.О тенденциях развития современных ВС и компьютерных сетей.	;
3.О состоянии дел на рынке современных компьютеров (список TOP500).	;
4.Принципы построения и архитектуру ЭВМ, основные компоненты традиционных ЭВМ. Организацию памяти, способы управления памятью.	;
5.Функционирование процессора. Технику конвейеризации. Причины остановки конвейера. Способы уменьшения остановок конвейера.	;
6.Набор команд. Представление команды. Причины перехода к философии RISC. Основные характеристики RISC-процессоров.	;
7.Организацию архитектур с параллелизмом на уровне данных (векторно-конвейерные, матричные, ассоциативные).	;
8.Причины построения вычислительных систем (ВС). Классификацию ВС.	;
9.Вычислительные системы с разделяемой памятью (многопроцессорные ВС). Основные компоненты ВС. Программное обеспечение. Современные реализации мультипроцессорных ВС.	;
10.Компьютеры с распределенной памятью (многомашинные ВС). Основные компоненты ВС. Основные компоненты ВС. Коммуникационных сред. Кластеры. Программное обеспечение. Современные реализации многомашинных ВС.	;
.5. 3	
	,
	,
11.Производительность ВС. Анализ последней редакции списка TOP500.	

.9. / . 3	
12. Принципы построения и архитектура компьютерных сетей. Организацию протокола.	;
13. Способы передачи информации в сетях. методы маршрутизации, модемы и сетевые карты. Коммутирующие устройства. Программное обеспечение сетей.	;
14. Архитектуру локальных вычислительных сетей, топологию сетей связи.	;
.5. 1	
15. Выбрать архитектуру вычислительной системы для эффективного решения задачи в данной предметной области.	
.5. 2	
16. Написать и отладить программу с учетом особенностей архитектуры ВС (например, доступа к памяти).	;
.9. / . 3	
17. Подключать персональный компьютер к локальной сети.	;
18. Написать и отладить программу с учетом особенностей архитектуры ЭВМ (например, организации памяти и т.д.).	;
19. Разработки сетевых приложений с использованием программного интерфейса.	;
20. Использования стандартных программных средств исследования компьютерных сетей на базе протоколов семейства TCP/IP.	;

3.

3.1

: 2				
	:		,	.
1.	:			
	1	3	1, 2, 3	
(,	,	,)
	:	.		

2.	0,5	3	4, 5, 6	
: RISC-				
3. CISC- RISC- RISC- CISC RISC	0	2	4, 5, 6	
:				
4. DLP- (Cray XT5 Quad Core, NEC SX-9). SIMD-	0	3	7, 8	
:				
5.	1	5	10, 11, 15, 16, 18, 8, 9	
:				
6.	1	4	10, 11, 15, 16, 18, 8, 9	
:				

<p>7.</p> <p>(MIPS, Flops).</p> <p>LINPAK, SPEC</p>	0	1	11	
<p>;</p>				
<p>8.</p> <p>OSI. OSI</p> <p>((MAC-IP, IP-), TCP</p> <p>UDP), TCP/IP,</p> <p>OSI.</p>	1	7	12, 13, 14, 19, 20	
<p>;</p>				
<p>9.</p> <p>().</p>	1	7	14, 17	

:				
10.	0,5	1	11, 13, 14	

4.

: 2				
1		16, 18, 4	10	2
: .]; - . - . , 2014. - 147, [1] . : . , . - . : http://elibrary.nstu.ru/source?bib_id=vtls000190282				
2		1, 10, 12, 13, 14, 2, 3, 4, 5, 8, 9	0	0
: .]; - . - . , 2014. - 147, [1] . : . , . - . : http://elibrary.nstu.ru/source?bib_id=vtls000190282				
3		12, 13, 14, 17, 19, 2, 20	0	0
: .]; - . - . , 2014. - 147, [1] . : . , . - . : http://elibrary.nstu.ru/source?bib_id=vtls000190282				
4		1, 10, 12, 13, 14, 2, 3, 4, 5, 6, 7, 8, 9	20	2
: .]; - . - . , 2014. - 147, [1] . : . , . - . : http://elibrary.nstu.ru/source?bib_id=vtls000190282				

5.

- , (. 5.1).

5.1

	-
	e-mail;
	e-mail;
	e-mail
	e-mail; ;

6.

(),

. 6.1.

- 15-

ECTS.

: 2		
<i>РГЗ №1:</i> Измерение времени работы программы по выбранному алгоритму. Исследование влияния опций компилятора на время выполнения программы.	10	20
" ... 2014. - 147, [1] ... : http://elibrary.nstu.ru/source?bib_id=vtls000190282"		
<i>РГЗ №2:</i> Исследование влияния параметров иерархии памяти на время выполнения программы.	15	30
" ... 2014. - 147, [1] ... : http://elibrary.nstu.ru/source?bib_id=vtls000190282"		
<i>РГЗ №3:</i> Исследование векторных расширений на простейших алгоритмах	15	30
" ... 2014. - 147, [1] ... : http://elibrary.nstu.ru/source?bib_id=vtls000190282"		
<i>Зачет:</i>	10	20
" ... 2014. - 147, [1] ... : http://elibrary.nstu.ru/source?bib_id=vtls000190282"		
<i>Зачет:</i>	0	20

6.2

.5	3.		+
	1.		+
	2.	+	+
	.9. / 3.	+	+

1

7.

1. Орлов С. А. Организация ЭВМ и систем : [учебник для вузов] / С. А. Орлов, Б. Я. Цилькер. - СПб. [и др.], 2011. - 686 с. : ил., табл.

2. Новожилов О. П. Архитектура ЭВМ и систем : учебное пособие для бакалавров / О. П. Новожилов. - Москва, 2012. - 527 с. : ил., табл.

3. Жмакин А. П. Архитектура ЭВМ : [учебное пособие] / Анатолий Жмакин. - Санкт-Петербург, 2010. - 344 с. : ил., табл. + 1 CD-ROM. - На обл. в подзаг.: Функциональная организация ЭВМ, машинная арифметика и синтез устройств, программные модели АЛУ, архитектура микропроцессорных систем, программная модель учебной ЭВМ, лабораторный практикум и курсовое проектирование.
4. Матушкин Г. Г. Архитектура компьютера [Электронный ресурс] : учебное пособие / Г. Г. Матушкин ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000166425. - Загл. с экрана.

1. Корнеев В. В. Современные микропроцессоры / Виктор Корнеев, Андрей Киселев. - СПб., 2003. - 440 с. : ил.
2. Степанов А. Н. Архитектура вычислительных систем и компьютерных сетей : [учебное пособие для вузов по специальности "Прикладная математика и информатика" (010200) и по направлению "Прикладная математика и информатика" (510200)] / А. Н. Степанов. - СПб. [и др.], 2007. - 508 с. : ил., табл.. - Издательская программа 300 лучших учебников для высшей школы.
3. Олифер В. Г. Компьютерные сети. Принципы, технологии, протоколы : учебное пособие для вузов / В. Г. Олифер, Н. А. Олифер. - СПб. [и др.], 2007. - 957 с. : ил.
4. Бройдо В. Л. Вычислительные системы, сети и телекоммуникации : учебное пособие для вузов по направлениям "Прикладная информатика" и "Информационные системы в экономике" / В. Л. Бройдо. - М. [и др.], 2006. - 702 с. : ил.
5. Таненбаум Э. С. Архитектура компьютера : [перевод] / Э. Таненбаум. - СПб., 2006. - 698 с. : ил.
6. Корнеев В. В. Вычислительные системы / В. В. Корнеев. - М., 2004. - 510, [1] с. : ил.
7. Таненбаум Э. С. Компьютерные сети : [пер. с англ.] / Э. Таненбаум. - СПб. [и др.], 2007. - 991 с. : ил.
8. Касперски К. Техника оптимизации программ. Эффективное использование памяти / Крис Касперски. - СПб., 2003. - 455, [1] с. : ил. + 1 CD-ROM.
9. Бройдо В. Л. Архитектура ЭВМ и систем : учебник для вузов по специальности "Информационные системы" / В. Л. Бройдо, О. П. Ильина. - СПб. [и др.], 2006. - 717 с. : ил.. - Издательская программа 300 лучших учебников для высшей школы в честь 300-летия Санкт-Петербурга.

1. ЭБС НГТУ : <http://elibrary.nstu.ru/>
2. ЭБС «Издательство Лань» : <https://e.lanbook.com/>
3. ЭБС IPRbooks : <http://www.iprbookshop.ru/>
4. ЭБС "Znaniy.com" : <http://znaniy.com/>
5. :

8.

8.1

1. Эффективное программирование современных микропроцессоров : учебное пособие / [В. П. Маркова и др.] ; Новосиб. гос. техн. ун-т. - Новосибирск, 2014. - 147, [1] с. : ил., табл.. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000190282

8.2

1 GCC

9. -

1	(-) , ,	

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»

Кафедра параллельных вычислительных технологий
Кафедра теоретической и прикладной информатики

“УТВЕРЖДАЮ”
ДЕКАН ФПМИ
д.т.н., доцент В.С. Тимофеев
“ ____ ” _____ г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

УЧЕБНОЙ ДИСЦИПЛИНЫ

Архитектура вычислительных систем и компьютерных сетей

Образовательная программа: 02.03.03 Математическое обеспечение и администрирование информационных систем, профиль: Математическое и программное обеспечение информационных технологий

1. **Обобщенная структура фонда оценочных средств учебной дисциплины**

Обобщенная структура фонда оценочных средств по дисциплине **Архитектура вычислительных систем и компьютерных сетей** приведена в Таблице.

Таблица

Формируемые компетенции	Показатели сформированности компетенций (знания, умения, навыки)	Темы	Этапы оценки компетенций	
			Мероприятия текущего контроля (курсовой проект, РГЗ(Р) и др.)	Промежуточная аттестация (экзамен, зачет)
ОПК.5 владение информацией о направлениях развития компьютеров с традиционной (нетрадиционной) архитектурой; о тенденциях развития функций и архитектур проблемно-ориентированных программных систем и комплексов	з3. Знать терминологию, принципы организации ЭВМ и ВС, их характеристики, узкие места и пути увеличения производительности	Пиковая и реальная производительность. Единицы измерения (MIPS, Flops). Закон Амдала. Способы измерения реальной производительности. Тесты LINPAK, SPEC и другие. Тенденции развития компьютерных сетей.	РГЗ, разделы 1-5 задания 1, разделы 1-3 задания 2, разделы 1-2 задания 3.	Зачет, вопросы 10-19
ОПК.5	у1. Уметь выявлять узкие места в прикладных программах и оптимизировать программы под заданную целевую архитектуру ЭВМ и вычислительных систем	Принципы построения вычислительных сред. Соединительные и функциональные элементы среды. Универсальность элементов вычислительной среды. Настройка среды. технология распределенной обработки данных; Физическая реализация элементов вычислительной среды. ПЛИСы. Способы организации ВС. Классификация ВС. Многомашинные ВС . Многопроцессорные ВС Однородные вычислительные системы. Поддержка когерентности кэш-памяти. Коммуникационные среды. Кластеры. Программное обеспечение ВС. Технология распределенной обработки данных. Развитие современных многопроцессорных и многомашинных ВС.		Зачет, вопросы 1-11, 20
ОПК.5	у2. Владеть навыками практической работы на компьютерах разной архитектуры	Принципы построения вычислительных сред. Соединительные и функциональные элементы среды. Универсальность элементов вычислительной среды. Настройка среды. технология распределенной обработки данных; Физическая реализация		Зачет, вопросы 1-11, 20

		<p>элементов вычислительной среды. ПЛИСы. Способы организации ВС. Классификация ВС. Многомашинные ВС . Многопроцессорные ВС Однородные вычислительные системы. Поддержка когерентности кэш-памяти. Коммуникационные среды. Кластеры. Программное обеспечение ВС. Технология распределенной обработки данных. Развитие современных многопроцессорных и многомашинных ВС.</p>		
<p>ПК.9.В/ОУ способность к выбору архитектуры и комплексирования современных компьютеров, систем, комплексов и сетей</p>	<p>з3. Знать архитектуру современных вычислительных систем и компьютерных систем</p>	<p>CISC-архитектуры. Формирование концепции RISC-архитектуры. Основные характеристики RISC-архитектур. Понятие регистрового окна. Оптимизирующий компилятор. Сравнительный анализ CISC и RISC архитектур. Принципы построения и архитектура компьютерных сетей. Многоуровневое сетевое взаимодействие. Понятие протокола, режимы его работы: соединение, передача сообщений, разъединение. Передача сообщений. Эталонная модель OSI. Уровни модели OSI и их функции: физический, канальный (уровень передачи данных) (MAC-адреса), сетевой (протокол IP, IP-адреса), транспортный (протоколы TCP и UDP), сеансовый, уровень предоставления данных, прикладной. Эталонная модель межсетевых протоколов TCP/IP, сравнение с OSI. Передача информации в компьютерных сетях; каналы связи, модемы. Структура пакета. Маршрутизация: методы маршрутизации, модемы и сетевые карты. Коммутирующие устройства. Типы коммутации: коммутации каналов, сообщений, пакетов. Программное обеспечение компьютерных сетей. Проблемы секретности в сетях ЭВМ и методы криптографии. Процессор, состав и функционирование. Набор команд. Техника конвейеризации. Операционный конвейер.</p>	<p>РГЗ, разделы разделы 1-2 задания 3</p>	<p>Зачет, вопросы 3-6, 9-15</p>

		Причины приостановки конвейера и техника их преодоления. Способы организации ВС. Классификация ВС. Многомашинные ВС . Многопроцессорные ВС Однородные вычислительные системы. Поддержка когерентности кэш-памяти. Коммуникационные среды. Кластеры. Программное обеспечение ВС. Технология распределенной обработки данных. Развитие современных многопроцессорных и многомашинных ВС.		
--	--	--	--	--

2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по дисциплине проводится в 2 семестре - в форме дифференцированного зачета, который направлен на оценку сформированности компетенций ОПК.5, ПК.9.В/ОУ. Из времени на самостоятельную работу на подготовку к промежуточной аттестации выделяется 20 часов.

Зачет проводится в устной форме, по билетам, составленным из вопросов, приведенных в паспорте зачета, позволяющих оценить показатели сформированности соответствующих компетенций

Кроме того, сформированность компетенций проверяется при проведении мероприятий текущего контроля, указанных в таблице раздела 1.

В 2 семестре обязательным этапом текущей аттестации является расчетно-графическое задание (работа) (РГЗ(Р)). Требования к выполнению РГЗ(Р), состав и правила оценки сформулированы в паспорте РГЗ(Р).

Общие правила выставления оценки по дисциплине определяются балльно-рейтинговой системой, приведенной в рабочей программе дисциплины.

На основании приведенных далее критериев можно сделать общий вывод о сформированности компетенций ОПК.5, ПК.9.В/ОУ, за которые отвечает дисциплина, на разных уровнях.

Общая характеристика уровней освоения компетенций.

Ниже порогового. Уровень выполнения работ не отвечает большинству основных требований, теоретическое содержание курса освоено частично, пробелы могут носить существенный характер, необходимые практические навыки работы с освоенным материалом сформированы не достаточно, большинство предусмотренных программой обучения учебных заданий не выполнены или выполнены с существенными ошибками.

Пороговый. Уровень выполнения работ отвечает большинству основных требований, теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые виды заданий выполнены с ошибками.

Базовый. Уровень выполнения работ отвечает всем основным требованиям, теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые из выполненных заданий, возможно, содержат ошибки.

Продвинутый. Уровень выполнения работ отвечает всем требованиям, теоретическое содержание

курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Новосибирский государственный технический университет»
Кафедра параллельных вычислительных технологий
Кафедра теоретической и прикладной информатики

Паспорт зачета

по дисциплине «Архитектура вычислительных систем и компьютерных сетей», 2 семестр

1. Методика оценки

Зачет проводится в устной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-10, второй вопрос из диапазона вопросов 11-20 (список вопросов приведен ниже). В ходе экзамена преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

Форма билета для зачета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет ФПМИ

Билет № _____

к зачету по дисциплине «Архитектура электронно-вычислительных машин и
вычислительных систем»

Вопрос 1 Кэш-память. Принцип построения кэш-памяти. Алгоритмы замещения. Проблемы поддержания когерентности. Понятие виртуальной памяти.

Вопрос 2. Топология локальных вычислительных сетей конфигурация связей. Устройства межсетевого интерфейса. Программное обеспечение компьютерных сетей.

Утверждаю: зав. кафедрой _____ профессор, Малышкин В.Э.
(подпись)

(дата)

2. Критерии оценки

- Ответ на билет для зачета считается **неудовлетворительным**, если студент при ответе на вопросы не дает определений основных понятий, не способен показать причинно-следственные связи явлений, в ответах на вопросы допускает принципиальные ошибки, оценка составляет от 0 до 9 баллов.
- Ответ на билет для зачета засчитывается на **пороговом** уровне, если студент при ответе на вопросы дает определение основных понятий, может показать причинно-следственные связи явлений, в ответах на вопросы допускает непринципиальные ошибки, оценка составляет от 10 до 15 баллов.
- Ответ на билет для зачета билет засчитывается на **базовом** уровне, если студент при

ответе на вопросы формулирует основные понятия, законы, дает характеристику процессов, явлений, проводит анализ причин, условий, может представить качественные характеристики процессов, не допускает серьезных ошибок в ответах на вопросы, оценка составляет от 16 до 17 баллов.

- Ответ на билет для зачета билет засчитывается на **продвинутом** уровне, если студент при ответе на вопросы проводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы решения, способен представить количественные характеристики определенных процессов, приводит конкретные примеры из практики, не допускает ошибок в ответах на вопросы, оценка составляет от 18 до 20 баллов.

3. Шкала оценки

- Зачет считается сданным, если сумма баллов по всем заданиям билета оставляет не менее 10 баллов (из 20 возможных).
- В общей оценке по дисциплине баллы за зачет учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. Вопросы к зачету по дисциплине «Архитектура вычислительных систем и компьютерных сетей»

Вопрос 1. Основные архитектурные принципы построения компьютера. Компьютер фон Неймана. Узкие места компьютера фон Неймана и его усовершенствования.

Вопрос 2. Иерархия памяти. Основной принцип построения иерархической памяти. Типичная схема иерархии памяти. Банки памяти. Интерливинг.

Вопрос 3. Кэш-память. Принцип построения кэш-памяти. Алгоритмы замещения. Проблемы поддержания когерентности. Понятие виртуальной памяти.

Вопрос 4. Процессор, состав и функционирование. Набор команд. Конвейерная обработка. Причины приостановки конвейера и техника их преодоления.

Вопрос 5. Архитектура с сокращенным набором команд (RISC). Основные характеристики RISC-процессоров. Конвейер RISC-процессоров. Понятие регистрового окна. Оптимизирующий компилятор.

Вопрос 6. Понятие последовательного и параллельного исполнения. Уровни параллелизма. Классификация Флинна. Уточненная классификация. Идея SIMD-расширений. Форматы SIMD-команд.

Вопрос 7. Классификация архитектур с параллелизмом на уровне данных. Архитектура суперскалярных микропроцессоров. Базисные принципы организации векторных, векторно-конвейерных, ассоциативных и матричных процессоров.

Вопрос 8. Принципы построения вычислительных систем (параллельных компьютеров). Классификация ВС. Компьютеры с разделяемой памятью (UMA, NUMA, cc-NUMA,) (мультипроцессоры). Программное обеспечение. Примеры современных мультипроцессоров, области применения.

Вопрос 9. Мультиядерные архитектуры. Понятие треда. Варианты реализации многопоточности. Примеры современных многоядерных процессоров, области применения.

Вопрос 10. Компьютеры с распределенной памятью (мультикомпьютеры). Структура мультипроцессора и мультикомпьютера. Поддержка когерентности кэш-памяти. Коммуникационные среды (SCI, Fast Ethernet, Gigabit Ethernet Myrinet, Infiniband, Quadrics). Программное обеспечение. Примеры современных мультикомпьютеров, области применения.

Вопрос 11. Кластеры. Коммуникационные среды. Программное обеспечение (OpenMP, MPI, треды). Технология распределенной обработки данных. Примеры современных

кластеров, области применения.

Вопрос 12. Принципы построения и архитектура компьютерных сетей. Многоуровневое сетевое взаимодействие. Понятие протокола. режимы их работы: соединение, передача сообщений, разъединение. Передача сообщений.

Вопрос 13. Эталонная модель OSI. Уровни модели OSI и их функции: физический, канальный (уровень передачи данных), сетевой (протокол IP, IP-адреса), транспортный (протоколы TCP и UDP), сеансовый, уровень предоставления данных, прикладной.

Вопрос 14. Протоколов TCP/IP. Передача информации в компьютерных сетях; каналы связи, модемы. Структура пакета.

Вопрос 15. Маршрутизация: методы маршрутизации, модемы и сетевые карты. Коммутирующие устройства. Типы коммутации: коммутации каналов, сообщений, пакетов. Программное обеспечение компьютерных сетей.

Вопрос 16. Принципы построения и архитектура локальные вычислительные сети; структура и принципы построения ЛВС;

Вопрос 17. Топология локальных вычислительных сетей конфигурация связей. Устройства межсетевого интерфейса. Программное обеспечение компьютерных сетей.

Вопрос 18. Примеры современных компьютерных и локальных сетей.

Вопрос 19. Анализ списка TOP500. Тенденции развития микропроцессоров и вычислительных систем. Оценки производительности.

Вопрос 20. Инструментальное программное обеспечение программиста. Оптимизирующий компилятор. Компоновщик. Отладчик и профилировщик. Назначение и основные функции.

Паспорт расчетно-графического задания (работы)

по дисциплине «Архитектура вычислительных систем и компьютерных сетей», 2 семестр

1. Методика оценки

В рамках расчетно-графического задания (работы) по дисциплине студенты должны построить программную реализацию для задачи по заданному варианту.

При выполнении расчетно-графического задания (работы) студенты должны выбрать алгоритм решения задачи по заданному варианту, построить его программную реализацию, оптимизировать ее и собрать статистику времени исполнения.

Отчет должен содержать титульный лист, описание целей работы, листинги программ, таблицы и графики с собранной статистикой и выводы по проделанной работе.

На оценку работы влияет соответствие реализованного набора функций исходному описанию задачи, качество построенной реализации (в первую очередь надежность и эффективность), полнота собранной статистики времени выполнения, а также качество оформления полученных результатов в виде отчета по РГР.

2. Критерии оценки

- Работа считается **не выполненной**, если отсутствует работающая реализация задачи по выбранному варианту, оценка составляет от 0 до 39 баллов.
- Работа считается выполненной **на пороговом** уровне, если реализация для задачи по выбранному варианту написана не полностью и с недочетами, оценка составляет от 40 до 57 баллов.
- Работа считается выполненной **на базовом** уровне, если полностью написана, но не оптимизирована по времени исполнения реализация для задачи по выбранному варианту, собрана не вся необходимая статистика, оценка составляет от 58 до 69 баллов.
- Работа считается выполненной **на продвинутом** уровне, если полностью написана и оптимизирована по времени исполнения реализация для задачи по выбранному варианту, собрана вся необходимая статистика, оценка составляет от 70 до 80 баллов.

3. Шкала оценки

- РГЗ считается сданным, если выставленные баллы составляют не менее 40 баллов (из 80 возможных).
- В общей оценке по дисциплине баллы за зачет учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. Примерный перечень тем РГЗ(Р)

Задание №1. Определение времени работы прикладных программ, исследование зависимости времени выполнения программ от уровня оптимизации компилятора.

Цели работы:

1. Научиться измерять интервалы времени в программах на языке С.

2. Исследовать зависимость времени работы прикладных программ от уровня оптимизации компилятора.

Форма отчета:

Результаты работы представляются в виде файла с отчётом о проделанной работе в произвольной форме. Отчёт должен содержать следующую информацию:

1. Исполнитель (ФИО, № группы)
2. Вариант задания
3. Таблица измерений, содержащая следующие поля:
 - Уровень оптимизации
 - Способ измерения времени
 - Величина параметра прикладного алгоритма
 - Время работы функции (с указанием единиц измерений)
4. Команды компиляции и запуска (с учётом регистра символов)
5. Исходный код программы

Задание:

1. Реализовать на языке C или C++ одну из функций на выбор (см. след. раздел). Проверить правильность работы программы.
2. Измерить время работы вычислительной функции полученной программы с использованием трёх различных способов измерения времени из приложения Б. При этом необходимо придерживаться методики измерения времени.
3. Параметр прикладного алгоритма необходимо выбрать таким образом, чтобы относительная погрешность измерения времени не превышала 1%.
4. Реализованную в п. 1 программу скомпилировать компилятором GNU C/C++ Compiler (далее GCC) с уровнями оптимизации -O0, -O1, -O2, -O3 (регистр ключей имеет значение), а также под архитектуру процессора, на котором будет осуществляться запуск программы. Измерить время работы программы при фиксированном параметре – одном и том же значении N для всех уровней.
5. Объяснить зависимость времени работы программы от уровня оптимизации.

Варианты функций:

- Вычисление числа Пи с помощью разложения в ряд по формуле:

$$\pi = 4 - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \dots,$$

параметр N – число итераций.

- Вычисление определенного интеграла сложной функции методом трапеций:

$$\int_a^b f(x) dx = \frac{b-a}{N} \sum_{k=0}^{N-1} \frac{f(x_k) + f(x_{k+1})}{2},$$

$f(x) = e^x \sin(x)$, $a = 0$, $b = \pi$.

Параметр: N - число интервалов.

- Вычисление квадратного корня с помощью алгоритма Ньютона:
Сходящаяся серия итераций: $a_{i+1} = (a_i + x / a_i) / 2$.

Начальное значение a_0 можно взять произвольным.

Параметр: N - число итераций.

- Сортировка методом пузырька.

Дан массив случайных чисел длины N . На первой итерации попарно упорядочиваются все соседние элементы; на второй - все, кроме последнего; на третьей - кроме последнего и предпоследнего и т.п.

Параметр: N - размер массива.

- Вычисление числа Пи методом Монте-Карло.

Генерируется N точек $(x;y)$, равномерно распределенных на квадрате $[0;1] \times [0;1]$. Вычисляется M - число точек, попавших в четверть круга с радиусом 1 ($x^2+y^2 \leq 1$). Число Пи можно приближенно вычислить по формуле:

$$\pi \approx \frac{4M}{N}.$$

Параметр: N - число точек.

Контрольные вопросы:

1. Почему при многократном измерении времени работы программы наиболее точным является минимальное время, а не среднее?
2. Почему не существует одного универсального способа измерения времени?
3. Назовите способы измерения времени работы программы. Перечислите их особенности.
4. Каким способом лучше измерять большие промежутки времени (порядка нескольких часов)?
5. Каким способом лучше измерять малые промежутки времени (порядка времени работы нескольких команд процессора)?
6. Всегда ли оптимизирующая компиляция позволяет уменьшить время работы программы?
7. Чем отличается общая оптимизация от оптимизации под архитектуру?

Задание №2. Исследовать влияние порядка обхода данных в кэш-памяти на время работы программы.

Цели работы:

1. Исследовать влияние кэш-памяти на время работы программ обработки данных.
2. Исследовать влияние порядка обработки данных в памяти на время работы программы.
3. Экспериментально определить некоторые параметры кэш-памяти (объем, размер строки и т.д.)

Задание:

1. Написать программу, многократно выполняющую чтение элементов массива заданного размера. Элементы массива представляют собой связный список, в котором значение очередного элемента представляет собой номер следующего. Способы обхода массива — прямой, обратный и случайный.

2. Построить графики зависимости среднего времени обращения к элементу массива от размера обрабатываемого массива для трех видов обхода. На графиках должно быть видно влияние кэш-памяти (1 и 2 уровня).

3. По результатам измерений сделать вывод о скорости различных способов обхода массива, а также о размерах различных уровней кэш-памяти (насколько полученные результаты соответствуют действительности).

Форма отчета

Результаты работы представляются в виде файла с отчетом о проделанной работе в произвольной форме. Отчет должен содержать следующую информацию:

1. Исполнитель (ФИО, № группы)
2. График зависимостей среднего времени доступа к одному элементу массива от общего размера массива (для 3-х вариантов обхода, на одном графике).
3. Команды компиляции и запуска программы
4. Предположение о размерах кэша 1-го и 2-го уровней (на основе графика)
5. Исходный код программы

Контрольные вопросы:

1. Объясните, почему случайный обход массива выполняется, как правило, дольше, чем последовательные.
2. Почему для малых размеров массива все три способа обхода массива работают с одинаковой скоростью?
3. Как будет вести себя график, если его неограниченно продолжить в сторону увеличения размера массива?

Задание №3 «Использование векторных расширений»

Цели работы:

1. Научиться использовать в программах на языке C векторные расширения архитектуры x86.
2. Исследовать уменьшение времени работы программы при векторизации вычислений.

Задание:

Обращение матрицы A размером N*N можно выполнить с помощью разложения в ряд:

$$A^{-1} = (I + R + R^2 + \dots)B,$$

где $R = I - BA$,

$$B = \frac{A^T}{\|A\|_1 \cdot \|A\|_\infty}, \quad \|A\|_1 = \max_j \sum_i |A_{ij}|, \quad \|A\|_\infty = \max_i \sum_j |A_{ij}|,$$

I - единичная матрица (на диагонали - единицы, остальные - нули), A_{ij} - элемент матрицы A с индексом (i,j).

- 1) Написать три варианта программы вычисления обратной матрицы:

- без использования специальных расширений (обычный вариант),
- с использованием встроенных векторных функций расширения SSE.
- с использованием библиотеки BLAS.

Каждый вариант программы:

- оптимизировать по скорости, насколько это возможно,
- проверить на правильность на небольшом тесте.

Использовать тип данных float.

Размер N предполагать кратным четырем.

2) Сравнить время работы четырех вариантов программы для $N=1024$, число шагов 10:

- обычный без векторизации компилятором,
- обычный с векторизацией компилятором (нужно указать тип процессора),
- с ручной векторизацией,
- с использованием BLAS.

Определить:

- среднее время одной итерации цикла (умножение + сложение матриц),
- время вычислений вне цикла (общее время минус время в цикле).

Замер времени выполнить несколько раз, в качестве результата взять минимальное время. Для измерения времени использовать функцию `times` (измерения времени работы процесса).

По результатам измерений сделать вывод.

Контрольные вопросы:

1. Объясните основную идею векторизации вычислений.
2. Любую ли программу можно векторизовать?
3. Какие векторные регистры используются в расширении SSE?