

«

»

“ ”

“ ”

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ Суперкомпьютеры и системы

: 09.03.01

, :

: 4, : 7

| | | |
|-----------|---------|----------|
| | | |
| | | 7 |
| 1 | () | 3 |
| 2 | | 108 |
| 3 | , . | 47 |
| 4 | , . | 18 |
| 5 | , . | 0 |
| 6 | , . | 18 |
| 7 | , . | 8 |
| 8 | , . | 2 |
| 9 | , . | 9 |
| 10 | , . | 61 |
| 11 | (, ,) | . |
| 12 | | |

(): 09.03.01

5 12.01.2016 ., : 09.02.2016 .

: 1, ,

(): 09.03.01

, 6 20.06.2017

, 6 21.06.2017

:

,

:

,

:

.

1.

1.1

| | |
|--|--|
| Компетенция НГТУ: ПК.10.В/ПТ готовность к разработке компонентов аппаратно-программных комплексов и баз данных с использованием современных инструментальных средств и технологий программирования; в части следующих результатов обучения: | |
| 12. | |
| 16. | |
| 7. | |
| 11. | |
| 8. | |

2.

2.1

| | |
|--|--|
| | |
|--|--|

| | |
|---|---|
| .10. / . 16 | |
| 1.О проблемах и направлениях развития суперкомпьютеров. | ; |
| .10. / . 7 | |
| 2.Об архитектуре параллельных вычислительных систем. | ; |
| .10. / . 16 | |
| 3.Об аппаратном и программном обеспечении суперкомпьютеров. | ; |
| .10. / . 7 | |
| 4.Об основных методах и средствах проектирования, производства, испытаний и оценки качества параллельных вычислительных систем. | |
| .10. / . 12 | |
| 5.Методы распараллеливания задач из различных прикладных областей. | |
| .10. / . 7 | |
| 6.Основные аппаратные и программные компоненты архитектуры суперкомпьютеров и систем | |
| .10. / . 16 | |
| 7.Классификацию суперкомпьютеров и систем. | ; |
| .10. / . 7 | |
| 8.Методы оценки производительности вычислительных систем. | ; |
| .10. / . 12 | |
| 9.Основные компоненты системного программного обеспечения, концептуальные положения функционального и параллельного программирования, методы, способы и средства разработки программ в рамках этих направлений. | |
| 10.Методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного продукта. | |

| | |
|--|--|
| .10. / . 8 | |
| 11. Построения основных моделей информационных технологий и способов их применения для решения задач в предметных областях. | |
| .10. / . 11 | |
| 12. Практического распараллеливания алгоритмов реальных задач. | |
| .10. / . 12 | |
| 13. Модели параллельных алгоритмов и методы параллельного программирования. | |
| 14. Математические и алгоритмические модели, методы и инструментальные программные средства анализа и обработки экспериментальных данных на компьютерах. | |
| .10. / . 11 | |
| 15. Проектирования, реализации, оценки качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях. | |
| .10. / . 8 | |
| 16. Практической работы на современных вычислительных кластерах. | |

3.

3.1

| | | | | |
|-----|---|-----|---------|---|
| | | | | |
| : 7 | | | | |
| : | | | | |
| 1. | 0 | 0,5 | 1 | |
| : | | | | |
| 2. | 0 | 1 | 2, 3, 6 | <p>Beowulf.</p> <p>Regatta.</p> <p>BlueGene/P.</p> <p>IBM</p> |

| | | | | |
|----|---|---|------|------------------------------|
| 3. | 0 | 2 | 3,6 | - |
| 4. | 0 | 1 | 3,7 | - |
| 5. | 0 | 1 | 3,8 | , Top500 Top50. |
| : | | | | |
| 6. | 0 | 1 | 6,9 | , |
| 7. | 0 | 2 | 10,6 | , - - , BlueGene/P, |
| : | | | | |

| | | | | |
|-----|---|-----|-------|--|
| 11. | 0 | 2 | 10, 9 | |
| : | | | | |
| 12. | 0 | 0,5 | 1 | |

3.2

| | | | | |
|----|---|---|------------|---------|
| | | | | |
| :7 | | | | |
| : | | | | |
| 1. | 0 | 2 | 11, 7, 8 | Pelican |
| : | | | | |
| 2. | 2 | 4 | 11, 16 | Pelican |
| : | | | | |
| 3. | 4 | 8 | 12, 14, 15 | |
| : | | | | |
| 4. | 2 | 4 | 12, 16 | / |

4.

| | | | | |
|----|--|---|---|---|
| | | | | |
| :7 | | | | |
| 1 | | 2 | 8 | 0 |

| | | |
|---------------------|---|----|
| | | |
| : 7 | | |
| Лабораторная №1: | 6 | 12 |
| Лабораторная №2: | 6 | 12 |
| Лабораторная №3: | 6 | 12 |
| Лабораторная №4: | 6 | 12 |
| Контрольные работы: | 0 | 32 |
| Зачет: | 0 | 20 |

6.2

| | | | |
|--|------------|---|---|
| | | | |
| | | | |
| | .10. / 12. | | + |
| | .10. / 16. | + | + |
| | .10. / 7. | + | + |
| | .10. / 11. | | + |
| | .10. / 8. | | + |

1

7.

1. Малявко А. А. Суперкомпьютеры и системы [Электронный ресурс] : электронный учебно-методический комплекс / А. А. Малявко ; Новосиб. гос. техн. ун-т. - Новосибирск, [2015]. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000214817. - Загл. с экрана.
2. Малявко А. А. Программное обеспечение высокопроизводительных вычислений. Ч. 1 : конспект лекций / А. А. Малявко ; Новосиб. гос. техн. ун-т. - Новосибирск, 2013. - 97, [2] с. : ил., табл. - Режим доступа: http://elibrary.nstu.ru/source?bib_id=vtls000182343
3. Гергель В. П. Высокопроизводительные вычисления для многопроцессорных многоядерных систем : [учебник для вузов по направлениям ВПО 010400 "Прикладная математика и информатика" и 010300 "Фундаментальная информатика и информационные технологии"] / В. П. Гергель ; Б-ка Нижегор. гос. ун-та им. Н. И. Лобачевского. - М., 2010. - 539, [4] с. : ил., схемы, табл. - На обл. в вых. дан.: Суперкомпьютерный консорциум университетов России.
4. Хорошевский В. Г. Архитектура вычислительных систем : [учебное пособие для вузов по направлению "Информатика и вычислительная техника"] / В. Г. Хорошевский. - М., 2008. - 519 с. : ил., табл.

5. Орлов С. А. Организация ЭВМ и систем : [учебник для вузов] / С. А. Орлов, Б. Я. Цилькер. - СПб. [и др.], 2011. - 686 с. : ил., табл.
 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. Корнеев В. Д. Параллельное программирование кластеров : учебное пособие / В. Д. Корнеев ; Новосиб. гос. техн. ун-т. - Новосибирск, 2008. - 310, [1] с. : ил.
 9. 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.
-
1. 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.
 2. Пятибратов А. П. Вычислительные системы, сети и телекоммуникации : [учебник для вузов по специальности "Прикладная информатика в экономике"] / А. П. Пятибратов, Л. П. Гудыно, А. А. Кириченко ; под ред. А. П. Пятибратова. - М., 2006. - 558, [1] с. : схем.
 3. Мелехин В. Ф. Вычислительные машины, системы и сети : учебник : [для вузов по направлению подготовки бакалавров, магистров, специалистов "Автоматизация и управление"] / В. Ф. Мелехин, Е. Г. Павловский. - М., 2006. - 554, [1] с. : ил., табл.
 4. Олифер В. Г. Компьютерные сети. Принципы, технологии, протоколы : [учебное пособие для вузов по направлению "Информатика и вычислительная техника" и по специальностям "Вычислительные машины, комплексы, системы и сети", "Программное обеспечение вычислительной техники и автоматизированных систем"] / В. Олифер, Н. Олифер. - СПб. [и др.], 2010. - 943 с. : ил.
 5. Башлы П. Н. Современные сетевые технологии : учебное пособие для вузов по специальности 351400 - "Прикладная информатика (по областям)" и др. междисциплинарным специальностям / П. Н. Башлы. - М., 2006. - 334 с. : ил., табл.
 6. Пескова С. А. Сети и телекоммуникации : учебное пособие для вузов / С. А. Пескова, А. В. Кузин, А. Н. Волков. - М., 2009. - 349, [1] с. : ил.
 7. Велихов А. В. Компьютерные сети : учебное пособие для вузов по дисциплине "Компьютерные сети и телекоммуникации" / А. В. Велихов, К. С. Строчников, Б. К. Леонтьев. - М., 2004. - 319 с.
 8. Олифер В. Г. Сетевые операционные системы : учебное пособие для вузов по направлению подготовки дипломированных специалистов "Информатика и вычислительная техника" / В. Г. Олифер, Н. А. Олифер. - СПб. [и др.], 2007. - 538 с. : ил.
 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 "Znanium.com" : <http://znanium.com/>
5. :

8.

8.1

1. Быханов К. В. Вычислительные системы, сети и телекоммуникации: лабораторный практикум [Электронный ресурс] : учебно-методическое пособие / К. В. Быханов ; Новосиб. гос. техн. ун-т. - Новосибирск, [2007]. - Режим доступа:

http://ciu.nstu.ru/fulltext/unofficial/2012/lib_25381_1326163693.doc. - Загл. с экрана.

2. Бычков М. И. Лабораторный практикум "Вычислительные сети" и "Сетевые информационные технологии" [Электронный ресурс] : учебно-методическое пособие / М. И. Бычков ; Новосиб. гос. техн. ун-т. - Новосибирск, [2011]. - Режим доступа:

<http://ermak.cs.nstu.ru/metodvssitinf/lrvcsit.exe>. - Загл. с экрана.

8.2

1 Microsoft Windows

2 Microsoft Office

9.

-

| | | |
|---|-----------------|----------|
| | | |
| 1 | (Internet) | Internet |

1. Обобщенная структура фонда оценочных средств учебной дисциплины

Обобщенная структура фонда оценочных средств по дисциплине Суперкомпьютеры и системы 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) |
| ПК.10.В/ПТ готовность к разработке компонентов аппаратно-программных комплексов и баз данных с использованием современных инструментальных средств и технологий программирования | з7. знать современные суперкомпьютерные системы и принципы их построения | Архитектура суперкомпьютеров и систем Методы оценки производительности вычислительных систем. Основные элементы архитектур многопроцессорных вычислительных систем. Прикладное программное обеспечение Системное программное обеспечение Технологии параллельного программирования | Контрольные работы, темы 1 и 2 | Зачет, вопросы 1-7 |
| ПК.10.В/ПТ | з12. знать современные методы параллельного программирования | Администрирование суперкомпьютеров и систем Модели параллельных взаимодействующих процессов Прикладное программное обеспечение Синхронные и асинхронные параллельные вычислительные процессы Системное программное обеспечение Технологии параллельного программирования | | Зачет, вопросы 8-14 |
| ПК.10.В/ПТ | з16. знать области приложения суперкомпьютерных средств и технологий | Архитектура суперкомпьютеров и систем Введение в суперкомпьютерные технологии Заключение Классификация архитектур многопроцессорных вычислительных систем. Методы оценки производительности вычислительных систем. Основные элементы архитектур многопроцессорных вычислительных систем. | Контрольные работы, темы 3 и 4 | Зачет, вопросы 15 - 24 |
| ПК.10.В/ПТ | у8. уметь сравнивать и выбирать суперкомпьютерные средства и технологии под прикладную задачу | Программное обеспечение кластера | | Зачет, вопросы 25-32 |
| ПК.10.В/ПТ | у11. уметь распараллеливать алгоритмы и программы решения | Параллельное программирование для кластера | | Зачет, вопросы 33-38 |

| | | | | |
|--|------------------|--|--|--|
| | прикладных задач | | | |
|--|------------------|--|--|--|

2. Методика оценки этапов формирования компетенций в рамках дисциплины.

Промежуточная аттестация по дисциплине проводится в 7 семестре - в форме дифференцированного зачета, который направлен на оценку сформированности компетенций ПК.10.В/ПТ.

Зачет проводится в устной (письменной) форме, по билетам.

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

В 7 семестре обязательным этапом текущей аттестации является контрольная работа. Требования к выполнению контрольной работы, состав и правила оценки сформулированы в паспорте контрольной работы.

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 компетенции ПК.10.В/ПТ, 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.

Паспорт зачета

по дисциплине «Суперкомпьютеры и системы», 7 семестр

1. Методика оценки

Зачет проводится в устной форме, по билетам. Билет формируется по следующему правилу: первый вопрос выбирается из диапазона вопросов 1-19, второй вопрос из диапазона вопросов 20-38 (список вопросов приведен ниже). В ходе зачета преподаватель вправе задавать студенту дополнительные вопросы из общего перечня (п. 4).

Форма билета для зачета

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Факультет АВТФ

Билет № _____

к зачету по дисциплине «Суперкомпьютеры и системы»

1. Многопроцессорные вычислительные системы с общей и распределенной памятью.
2. Механизм "условных критических интервалов". Пример решения задачи "читатели-писатели" с помощью этого механизма.

Утверждаю: зав. кафедрой _____ должность, ФИО
(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 иллюстрациях к материалу допускает принципиальные ошибки, оценка составляет 0-9 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, поясняя основной материал, допускает непринципиальные ошибки, оценка составляет 10-14 баллов.
- Ответ на билет для зачета засчитывается на **базовом** уровне, если студент при ответе на вопросы четко формулирует основные понятия, законы, дает характеристику процессов, явлений, проводит анализ причин, условий, может представить качественные характеристики изучаемых процессов, оценка составляет 15-17 баллов.

- Ответ на билет для зачета засчитывается на **продвинутом** уровне, если студент при ответе на вопросы проводит сравнительный анализ подходов, проводит комплексный анализ, выявляет проблемы, предлагает механизмы их решения, способен представить количественные характеристики определенных процессов, приводит конкретные примеры из практики, оценка составляет 18-20 *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 «Суперкомпьютеры и системы»

1. Архитектура параллельных вычислительных систем. Классификация архитектур.
2. Параллельные и распределенные вычисления и их техническая основа - вычислительные кластеры, ГРИД-системы и суперкомпьютеры.
3. Виды параллелизма.
4. Основные проблемы параллельной обработки данных.
5. Закон Амдаля о предельном значении коэффициента ускорения за счет параллелизма.
6. Закон Мура о росте производительности последовательных компьютеров.
7. Закон Гроша о стоимости параллельных систем.
8. Гипотеза Минского о влиянии потерь на взаимодействие на степень ускорения параллельных вычислений по сравнению с последовательными.
9. Конвейерные и векторные вычисления. Процессорные матрицы.
10. Многопроцессорные вычислительные системы с общей и распределенной памятью.
11. Схемы коммутации.
12. Схемы взаимодействия ветвей параллельных алгоритмов
13. Основные компоненты аппаратного и программного обеспечения суперкомпьютеров.
14. Представление алгоритма. Определение реализации алгоритма, последовательного и параллельного представления алгоритма.
15. Определение сети Петри. Разметка сети Петри. Правила срабатывания переходов. Работа сети.
16. Граф достижимости сети Петри. Понятия ограниченной и безопасной сети. Понятие тупиковой разметки сети Петри.
17. Определение процесса. Два главных типа взаимодействия параллельных процессов. Задача взаимного исключения (определение).
18. Понятие критического интервала, разделяемого и неразделяемого ресурса. Пример сети Петри, моделирующей взаимное исключение.
19. Семафоры. Операции над семафорами. Пример сети Петри, моделирующей операции над семафорами.
20. Определение взаимной блокировки (дедлока). Необходимые условия дедлока. Пример сети Петри, допускающей дедлок.
21. Способы борьбы с дедлоками.
22. Механизм "условных критических интервалов". Пример решения задачи "читатели-писатели" с помощью этого механизма.
23. Монитор. Общее представление. Пример монитора для задачи "производитель-потребитель".
24. Модели параллельно-последовательного программирования. MPMD и SPMD модели программирования.

25. Параллельная программа разделения множеств (Дейкстры), идея доказательства её корректности.
26. Событийное управление. Типичные "локальные" ситуации, которые могут возникнуть в событийном управлении, представить сетью Петри.
27. Событийное управление. Операции над сетями Петри (присоединения, исключения, итераций, наложения, разметки), продемонстрировать примерами.
28. Потокное управление. Операции: преобразователь, синхронизатор, распределитель, селектор, арбитр. Примеры реализации управляющих конструкций этими операциями.
29. Потокное управление. Волновые вычисления. Пример волнового процессора умножения матрицы на матрицу.
30. Динамическое управление. Понятие программы в асинхронном динамическом программировании.
31. Вычислительная модель охраняемых команд. Вычислительная модель Ч.Хоара последовательных сообщающихся процессов.
32. Синхронные вычисления. Определение систолического вычислителя. Три фазы систолического алгоритма. Пример систолического процессора для умножения матрицы на вектор.
33. Основные директивы OpenMP.
34. Интерфейс MPI. Коммуникаторы.
35. Интерфейс MPI. Виды функций приема-передачи сообщений.
36. Интерфейс MPI. Индивидуальные взаимодействия.
37. Интерфейс MPI. Коллективные взаимодействия.
38. Интерфейс MPI. Односторонние взаимодействия.

Паспорт контрольной работы

по дисциплине «Суперкомпьютеры и системы», 7 семестр

1. Методика оценки

Контрольная работа проводится по 4 темам, каждая из которых включает от 14 до 32 заданий. Выполняется письменно. Студент получает по одному заданию из каждой темы.

Образец набора заданий на контрольную работу:

Задание 1. Что такое командная оболочка ОС Linux? Какие функции она выполняет?

Задание 2. Что такое неявная барьерная синхронизация в OpenMP? С помощью каких средств ее можно отменить?

Задание 3. Перечислите виды и характеристики памяти GPU, доступной из программы основного процессора.

Задание 4. Что такое коллективное взаимодействие ветвей программы в MPI?

2. Критерии оценки

Каждое задание контрольной работы оценивается в соответствии с приведенными ниже критериями.

Контрольная работа считается **невыполненной**, если студент не дал правильных ответов на три или все четыре задания. Оценка составляет 0-15 баллов.

Работа считается выполненной на **пороговом** уровне, если правильные ответы (возможно – с небольшими неточностями) даны на два задания из четырех. Оценка составляет 16-20 баллов.

Работа считается выполненной на **базовом** уровне, если правильные ответы (возможно – с небольшими неточностями) даны на три задания из четырех. Оценка составляет 21-25 баллов.

Работа считается выполненной на **продвинутом** уровне, если правильные ответы (возможно – с небольшими неточностями) даны на все четыре задания. Оценка составляет 26-32 баллов.

3. Шкала оценки

В общей оценке по дисциплине баллы за контрольную работу учитываются в соответствии с правилами балльно-рейтинговой системы, приведенными в рабочей программе дисциплины.

4. Варианты заданий на контрольную работу

Тема 1. Программное обеспечение суперкомпьютеров. Администрирование

Задание 5. Что такое суперпользователь в ОС Linux? Какие команды позволяют работать от имени суперпользователя?

Задание 6. Как устроено дерево каталогов в системе Linux?

Задание 7. Какие ограничения на имена файлов и каталогов имеются в ОС Linux?

Задание 8. Что такое жесткие и символические ссылки на файлы в ОС Linux?

- Задание 9. Какие типы файлов существуют в ОС Linux?
- Задание 10. Как организована система управления правами доступа в ОС Linux?
- Задание 11. Что такое командная оболочка ОС Linux? Какие функции она выполняет?
- Задание 12. Что такое автозавершение имен команд, файлов и каталогов в оболочке bash?
- Задание 13. Что такое псевдонимы, встроенные команды и другие виды команд оболочки bash?
- Задание 14. Что такое перенаправление потоков ввода-вывода и конвейеры в ОС Linux?
- Задание 15. Перечислите основные команды работы с файлами и каталогами в ОС Linux.
- Задание 16. С помощью каких команд Linux можно изменять владельца, группу и права доступа к файлам и каталогам?
- Задание 17. Что такое переменные, параметры и подстановки, используемые в командах оболочки bash?
- Задание 18. Охарактеризуйте средства программирования скриптов оболочки bash.

Тема 2. Программирование суперкомпьютеров. Программирование систем с общей памятью

- Задание 1. Перечислите составные части технологии OpenMP.
- Задание 2. С помощью какой директивы (директив) создаются новые параллельные области программы в OpenMP?
- Задание 3. Что такое критическая секция программы в OpenMP?
- Задание 4. Какими способами можно установить нужное количество потоков для создания очередной параллельной области в OpenMP?
- Задание 5. Как обеспечить выполнение фрагмента параллельной области OpenMP только главным потоком?
- Задание 6. Какая опция директивы OpenMP for используется для указания способа распределения итераций цикла между потоками параллельной области?
- Задание 7. Что такое deadlock в OpenMP? Каким правилам нужно следовать, чтобы избежать возможности попадания параллельной программы в deadlock?
- Задание 8. Что такое сведение данных в OpenMP? Какие опции и в каких директивах используются для выполнения сведения?
- Задание 9. Что делает директива OpenMP threadprivate?
- Задание 10. Как обеспечить выполнение фрагмента параллельной области OpenMP потоком с максимальным номером в данной параллельной области?
- Задание 11. Для чего в OpenMP используется опция firstprivate? Чем она отличается от опций private и lastprivate?
- Задание 12. Что такое вложенная параллельная область программы в OpenMP? В каких случаях ее нельзя создать?
- Задание 13. Что такое неявная барьерная синхронизация в OpenMP? С помощью каких средств ее можно отменить?
- Задание 14. Для чего используются директивы OpenMP sections и section? Что делает каждая из этих директив?
- Задание 15. Перечислите все средства синхронизации потоков в OpenMP.
- Задание 16. Перечислите возможные способы распределения итераций цикла между потоками в OpenMP.
- Задание 17. Что делает директива OpenMP atomic?
- Задание 18. В чем состоит различие между общими и локальными переменными потока в OpenMP?
- Задание 19. С помощью каких средств можно ограничить глубину вложенности параллельных областей программы в OpenMP?
- Задание 20. От чего зависит равномерность загрузки процессоров/ядер системы с общей памятью?

- Задание 21. Каким образом функция, вызываемая из параллельной программы в OpenMP, может выяснять, в последовательной или параллельной области она выполняется?
- Задание 22. Как обеспечить выполнение фрагмента параллельной области OpenMP в точности одним потоком?
- Задание 23. Какое значение будет иметь переменная count в результате выполнения фрагмента параллельной программы OpenMP:
- ```
int count = 0;
#pragma omp parallel for
for(int i = 0; i < 1000000; i++){
count++;
}

```
- Задание 24. Могут ли два потока OpenMP одновременно захватить один множественный замок?
- Задание 25. Для чего может использоваться опция reduction в некоторых директивах OpenMP?
- Задание 26. Каково назначение директивы OpenMP single?
- Задание 27. Каким ограничениям должен удовлетворять цикл для того, чтобы к нему можно было применить директиву OpenMP for?
- Задание 28. Чем в OpenMP отличается простой замок от множественного?
- Задание 29. Перечислите механизмы синхронизации потоков в OpenMP.
- Задание 30. Перечислите способы распределения итераций цикла между потоками параллельной области OpenMP.

### Тема 3. Программирование суперкомпьютеров. Программирование вычислительных акселераторов

- Задание 1. Что такое ядро в терминологии CUDA?
- Задание 2. Как задается размерность и количество потоков ядра CUDA, выполняемого графическим процессором?
- Задание 3. Перечислите виды и характеристики памяти, доступной из программы GPU.
- Задание 4. Как указать требуемое размещение переменной в памяти GPU (регистровой, разделяемой, глобальной, памяти текстур, ...)?
- Задание 5. Перечислите виды и характеристики памяти, доступной из программы CPU при использовании архитектуры CUDA.
- Задание 6. Какова максимальная размерность блока CUDA в потоках?
- Задание 7. Перечислите встроенные векторные типы данных расширения CUDA языка C и объясните смысл их наименований.
- Задание 8. Как в программе для CPU обеспечить синхронизацию с программой для GPU?
- Задание 9. Если к одному CPU подключено несколько видеокарт NVidia архитектуры CUDA, то каким образом можно обеспечить их одновременную загрузку?
- Задание 10. Перечислите виды и характеристики памяти GPU, доступной из программы основного процессора.
- Задание 11. Какие ограничения наложены на функции, которые должны выполняться на GPU?
- Задание 12. Что такое мультипроцессор в терминологии CUDA?
- Задание 13. Какие ограничения наложены на функции, выполняемые в графическом процессоре?
- Задание 14. Как определить версию и технические характеристики графического процессора NVIDIA?
- Задание 15. Перечислите и охарактеризуйте группы библиотечных функций Run-time библиотеки CUDA.
- Задание 16. Что такое сетка, блок, поток в терминологии CUDA?

- Задание 17. Что такое тип данных dim3 CUDA?
- Задание 18. Какова максимальная размерность сетки CUDA в блоках?
- Задание 19. Перечислите и охарактеризуйте функции основных компонент программного обеспечения CUDA.
- Задание 20. Можно ли (и если да - то как) в программе для CPU выяснить, какую версию спецификации CUDA реализует видеокарта?
- Задание 21. Какую модель параллелизма реализует архитектура CUDA?
- Задание 22. Что делает препроцессор nvcc CUDA?
- Задание 23. Можно ли получить указатель на функцию, выполняемую графическим процессором?
- Задание 24. С помощью какого расширения синтаксиса запускается ядро CUDA?
- Задание 25. Перечислите основные группы функций CUDA и охарактеризуйте их назначение.
- Задание 26. Перечислите и охарактеризуйте дополнительные виды функций, существующие в расширении синтаксиса языка C для CUDA.
- Задание 27. Какие виды памяти доступны из программ для GPU?
- Задание 28. Как синхронизируются процессы в CPU и в GPU?
- Задание 29. Можно ли одновременно использовать технологии OpenMP и CUDA?
- Задание 30. Какие имена полей используются во встроенных векторных типах CUDA?
- Задание 31. С данными какого типа оперируют арифметико-логические устройства GPU CUDA?
- Задание 32. Как узнать максимальное количество потоков в блоке для видеокарты NVIDIA, подключенной к CPU?

#### **Тема 4. Программирование суперкомпьютеров. Программирование кластеров**

- Задание 1. Какие виды виртуальных топологий можно реализовывать с использованием MPI?
- Задание 2. Что такое барьерная синхронизация в MPI? Какие еще виды синхронизации существуют в стандарте MPI-1?
- Задание 3. Перечислите основные группы функций стандарта MPI-1.
- Задание 4. Может ли MPI-программа, выполняющаяся на многоядерном узле, использовать все его ядра и если да, то каким образом?
- Задание 5. Что такое пользовательская операция редукции в MPI?
- Задание 6. Чем различаются размер и протяженность типов данных MPI?
- Задание 7. Что такое буферизованная передача сообщений в MPI?
- Задание 8. Что такое коммуникатор в MPI? Какие проблемы решаются с использованием этого понятия?
- Задание 9. Сколько в MPI операций приема сообщений типа "точка-точка"? Перечислите и охарактеризуйте их.
- Задание 10. Что такое коллективное взаимодействие ветвей программы в MPI?
- Задание 11. Как в MPI путем вызова одной функции можно одновременно передать и принять блок данных?
- Задание 12. Перечислите предопределенные операции редукции данных MPI.
- Задание 13. Что такое виртуальная топология в MPI? Как она связана с физической топологией вычислительной сети/кластера/комплекса?
- Задание 14. Для чего в MPI-программах можно использовать производные типы данных?
- Задание 15. Что понимается под операциями сдвига и циклического сдвига данных в MPI? Какие функции обычно используются для их реализации?
- Задание 16. Что такое исключаящая редукция (MPI\_Exscan), как она выполняется?
- Задание 17. Может ли ветвь параллельной программы с помощью MPI передавать сообщения сама себе?
- Задание 18. Какие операции рассылки и сбора данных реализованы в MPI?

- Задание 19. Должны ли абсолютно все ветви параллельной MPI-программы принимать участие в коллективном взаимодействии?
- Задание 20. Сколько конструкторов производных типов реализовано в MPI? Перечислите и охарактеризуйте их.
- Задание 21. Чем различаются виртуальные топологии MPI "тор" и "решетка"?