

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Чернецкая Ирина Евгеньевна
Должность: Заведующий кафедрой
Дата подписания: 02.10.2023 15:30:57
Уникальный программный ключ:
bdf214c64d8a381b0782ea566b0dce05e3f5ea2d

МИНОБРНАУКИ РОССИИ

Юго-Западный государственный университет

УТВЕРЖДАЮ

Заведующий кафедрой

вычислительной техники

 И.Е.Чернецкая

« 01 » сентября 2023 г.

ОЦЕНОЧНЫЕ СРЕДСТВА

для текущего контроля успеваемости
и промежуточной аттестации обучающихся
по дисциплине

ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ

(наименование дисциплины)

09.04.01 Информатика и вычислительная техника, направленность
(профиль) «Элементы и устройства вычислительной техники и

информационных систем»

(код и наименование ОПОП ВО)

1. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ

1.1. Вопросы для собеседования

1. Раздел (тема) дисциплины «Введение»

1. Какие основные тенденции развития средств вычислительной техники влияют на эволюцию инструментальных средств параллельного программирования?
2. Какие современные классы аппаратного обеспечения могут быть запрограммированы с использованием инструментальных средств параллельного программирования?
3. На какой класс аппаратных средств ориентировано многопоточное программирование?
4. Какие современные аппаратные средства соответствуют векторному принципу параллельного программирования?
5. В чем ключевое отличие систем с общей и распределенной памятью?
6. В чем ключевое отличие вычислительных кластеров/суперкомпьютеров и грид систем?
7. Какие инструментальные средства применяются для параллельного программирования GPU?
8. Сформулируйте закон Мура и оцените его влияние на развитие средств вычислительной техники с параллельной архитектурой.
9. Сформулируйте закон Деннарда и оцените его влияние на развитие средств вычислительной техники с параллельной архитектурой.

2. Раздел (тема) дисциплины «Понятие производительности вычислительной системы и способы ее измерения»

1. Дайте определение реальной и пиковой производительности вычислительной системы. В чем их ключевые отличия друг от друга?
2. В каких единицах измеряется производительность вычислительных систем?
3. Что такое бенчмарки и как они связаны с оценкой реальной производительности вычислительной системы?
4. Какие современные бенчмарки вы знаете?
5. Совпадают ли оценки реальной производительности одной и той же вычислительной системы, полученные с использованием различных бенчмарков? В чем причина?

3. Раздел (тема) дисциплины «Закон Амдала»

1. Сформулируйте закон Амдала.
2. Чем лимитируется выигрыш от распараллеливания последовательной программы?
3. В чем отличие идеального и реального применения закона Амдала на практике?
4. Всегда ли оправдано повышение числа процессоров с целью параллельного исполнения программного кода? Существует ли оптимальное число процессоров?
5. Любая ли программа допускает эффективное распараллеливание под заданный класс вычислительных средств с параллельной архитектурой? В чем причины/ограничения?

4. Раздел (тема) дисциплины «Средства аппаратной поддержки параллельного исполнения»

1. Сколько конвейеров имеют современные процессоры и какие алгоритмические приемы параллельного программирования применяются для их использования?
2. Сколько ядер имеют современные процессоры и какие инструментальные средства параллельного программирования применяются для их использования?

3. В чем ключевые отличия систем симметричной многопроцессорности (SMP) и систем с неоднородным доступом в память (NUMA) с позиции параллельного программирования?
4. В чем отличие физических и логических ядер процессора?
5. В чем ключевые особенности технологии Hyper-Threading с позиции параллельного программирования?
6. Какие инструментальные средства применяются для программирования векторных расширений (MMX, SSE, AVX) системы команд современных процессоров?
7. Какие примитивы синхронизации параллельно работающих процессов вы знаете? Какие аппаратные средства и сервисы современных операционных систем обеспечивают их функционирование?
8. Какие инструментальные средства, предоставляемые операционной системой, применяются для многопоточного программирования?
9. Какие инструментальные средства применяются для программирования вычислительных кластеров и суперкомпьютеров? В чем их ключевые отличия по сравнению с многопоточным программированием многоядерных/многопроцессорных вычислительных систем?
10. Какие инструментальные средства применяются для программирования грид систем?
11. В чем заключается проблема быстродействия памяти в современных вычислительных системах?
12. Для чего предназначена кэш-память и какие приемы параллельного программирования используются для организации ее эффективного функционирования?

5. Раздел (тема) дисциплины «Профилирование и оптимизация программных средств»

1. Дайте определение понятия узких мест программы?
2. Какие участки последовательной программы следуют подвергать оптимизации и/или распараллеливанию в первую очередь?
3. Дайте определение профайлера. Какие современные профайлеры применяются для оценки эффективности исполнения программного кода?
4. Какие уровни оптимизации программного обеспечения существуют? На каком из них достигается максимальный выигрыш от оптимизации и/или распараллеливания?
5. Приведите примеры алгоритмических приемов, используемых для оптимизации параллельного выполнения программ на каждом из уровней оптимизации.
6. Какие современные компиляторы вы знаете? В чем ключевые отличия программного кода на языке ассемблера, генерируемого ими?

Шкала оценивания: 48-балльная.

Критерии оценивания

2 балла выставляется обучающемуся, если он принимает активное участие в беседе по большинству обсуждаемых вопросов (в том числе самых сложных); демонстрирует сформированную способность к диалогическому мышлению, проявляет уважение и интерес к иным мнениям; владеет глубокими (в том числе дополнительными) знаниями по существу обсуждаемых вопросов, ораторскими способностями и правилами ведения полемики; строит логичные, аргументированные, точные и лаконичные высказывания, сопровождаемые яркими примерами; легко и заинтересованно откликается на неожиданные ракурсы беседы; не нуждается в уточняющих и (или) дополнительных вопросах преподавателя.

1 балл выставляется обучающемуся, если он принимает участие в обсуждении не менее 50% дискуссионных вопросов; проявляет уважение и интерес к иным мнениям, доказательно и корректно защищает свое мнение; владеет хорошими знаниями вопросов, в обсуждении которых принимает участие; умеет не столько вести полемику, сколько

участвовать в ней; строит логичные, аргументированные высказывания, сопровождаемые подходящими примерами; не всегда откликается на неожиданные ракурсы беседы; не нуждается в уточняющих и (или) дополнительных вопросах преподавателя.

0 баллов (выставляется обучающемуся, если он не владеет содержанием обсуждаемых вопросов или допускает грубые ошибки; пассивен в обмене мнениями или вообще не участвует в дискуссии; затрудняется в построении монологического высказывания и (или) допускает ошибочные высказывания; постоянно нуждается в уточняющих и (или) дополнительных вопросах преподавателя.

2. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ

2.1. Банк вопросов и заданий в тестовой форме

1. Вопросы в закрытой форме

1. Запуск двух вычислительно-интенсивных потоков на двух физических ядрах процессора в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности

- около 2 раз
- 10-20%
- выигрыша не будет

2. Запуск двух вычислительно-интенсивных потоков на двух логических ядрах процессора в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности

- 10-20%
- около 2 раз
- выигрыша не будет

3. Запуск двух вычислительно-интенсивных потоков на одноядерном процессоре в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности

- выигрыша не будет
- около 2 раз
- 10-20%

4. Запуск двух потоков с большим числом обращений в память на двух физических ядрах процессора в условиях отсутствия прочих конфликтов способен дать выигрыш в реальной производительности

- выигрыша не будет
- около 2 раз
- 10-20%

5. Какие из задач наиболее неудобны для параллельных вычислений?

- сильносвязанные
- слабосвязанные
- набор независимых подзадач

6. Какой из типов оптимизации кода программы не относится к программной оптимизации?

- параллельная
- высокоуровневая
- алгоритмическая
- микроархитектурная

7. Какие задачи возможно эффективно решать с использованием грид-систем?

- слабосвязанные
- сильсвязанные
- все вышеперечисленные

8. Какие задачи возможно эффективно решать с использованием суперкомпьютеров?
- все вышеперечисленные
 - слабосвязанные
 - сильсвязанные
9. Программный интерфейс MPI применяется для
- программирования под кластеры и суперкомпьютеры
 - программирования под GPU
 - программирования под грид-системы
 - программирования векторных расширений
10. Программный интерфейс OpenMP применяется для
- многопоточного программирования
 - программирования под кластеры и суперкомпьютеры
 - программирования под GPU
 - программирования под грид-системы
 - программирования векторных расширений
11. Инструментарий CUDA применяется для
- программирования под GPU
 - программирования под кластеры и суперкомпьютеры
 - программирования под грид-системы
 - программирования векторных расширений
 - многопоточного программирования

2. Вопросы в открытой форме

1. Какой из типов оптимизации кода программы дает максимальный выигрыш?
- алгоритмическая
 - высокоуровневая
 - микроархитектурная
2. Вынос инвариантов цикла является примером оптимизации следующего типа
- высокоуровневая
 - алгоритмическая
 - микроархитектурная
3. Раскрутка циклов является примером оптимизации следующего типа
- высокоуровневая
 - алгоритмическая
 - микроархитектурная
4. Изменение алгоритмов работы программы является примером оптимизации следующего типа
- алгоритмическая
 - высокоуровневая
 - микроархитектурная
5. Приспособление программы к особенностям конкретного процессора является примером оптимизации следующего типа

- микроархитектурная
 - высокоуровневая
 - алгоритмическая
6. Согласно закону Амдала выигрыш от распараллеливания программы ограничивается
- ее последовательной частью
 - числом процессоров
 - числом потоков
 - особенностями микроархитектуры процессора
7. Какая из перечисленных технологий не относится к векторным расширениям системы команд процессора?
- AES
 - MMX
 - SSE
 - AVX
8. Код с плавающей точкой какого компилятора характеризуется наихудшим качеством оптимизации?
- Borland C++ Builder
 - Intel C++ Compiler
 - Microsoft Visual C++
 - GCC
9. Какие из перечисленных оптимизаций не может выполнять компилятор?
- распараллеливание кода под MPI
 - раскрутка циклов
 - ликвидация хвостовой рекурсии
 - удаление константных параметров подпрограмм
 - векторизация кода под AVX
10. Что из нижеперечисленного не является примитивом синхронизации потоков?
- дедлоки
 - мьютексы
 - семафоры
 - критические секции
 - барьеры
11. Целью профилирования программы является
- выявление кода, на выполнение которого тратится максимальное количество времени
 - анализ времени выполнения программы в целом
 - замена скалярных ассемблерных инструкций на векторные
 - выявление приемов высокоуровневой оптимизации, которые необходимо применить к коду в ходе последующего этапа оптимизации
3. *Установление правильной последовательности*
1. Какой из способов измерения времени работы программы является наиболее точным?
- с использованием подсчета числа тактов тактовой частоты

- с использованием замера системного времени
 - с использованием подсчета числа событий WM_TIMER
 - с использованием счетчиков, управляемых через MSR-регистры
2. Целью проведения оптимизации наиболее часто является
- повышение скорости обработки данных
 - снижение затрат памяти при обработке данных
 - снижение интенсивности работы с динамической памятью
 - векторизация кода
 - распараллеливание кода
3. Какие из перечисленных средств не используются для программирования в рамках концепции GPGPU?
- OpenMP
 - CUDA
 - Stream
 - OpenCL
 - OpenACC
4. Процессор исполняет программы в виде
- двоичного кода
 - команд языка Delphi
 - команд языка C++
 - команд языка ассемблера
5. Компилятор предназначен для
- преобразования программы из языка высокого уровня в объектный код
 - получения готового исполнимого файла из объектного кода
 - преобразования exe-файла в com-файл
 - проверки синтаксиса и семантики программы
6. Оптимизация программы - это
- структурные преобразования, уменьшающие время выполнения программы или ее фрагмента
 - опция компилятора
 - средство затруднения отладки
 - структурные преобразования, направленные на изменение алгоритма программы с целью его упрощения
7. Всегда ли компилятор проводит оптимизацию и распараллеливание кода?
- зависит от установки опций компилятора
 - всегда
 - иногда
8. На всех этапах разработки оправдано применение оптимизации и распараллеливания?
- нет
 - да
9. В современных вычислительных системах оперативная память работает
- существенно медленнее процессора

- существенно быстрее процессора
- примерно на одной скорости с процессором

10. Какое из расширений программной модели не присутствует в современных процессорах?

- 3DFX
- MMX
- 3DNow
- SSE
- AVX

11. В системе команд современных процессоров нет команд

- с 4 операндами
- без операндов
- с 1 операндом
- с 3 операндами

4. Установление соответствия

1. Команды целочисленного умножения и деления на аргумент, являющийся степенью двойки, рекомендуется заменять на:

- команды сдвига, т.к. они работают быстрее
- команды сложения, т.к. их обычно требуется меньше, чем сдвигов
- логические команды в соответствии с алгоритмом действия
- заменять не рекомендуется

2. Количество команд условных переходов необходимо по возможности минимизировать, т.к.

- плохопредсказываемые переходы ведут к частым сбросам конвейера
- команды условных переходов имеют существенную длину в машинном коде
- команды условных переходов имеют большую длительность исполнения
- переходы снижают читабельность кода

3. Расширение MMX предназначено

- для векторной обработки целочисленных данных
- для векторной обработки вещественных чисел
- для быстрой горизонтальной обработки символьной информации
- для автоматического распараллеливания вычислений

4. Расширение SSE предназначено

- для векторной обработки вещественных чисел
- для векторной обработки целочисленных данных
- для быстрой горизонтальной обработки символьной информации
- для автоматического распараллеливания вычислений

5. Расширение SSE2 предназначено

- для векторной обработки целочисленных и вещественных данных
- для векторной обработки только вещественных чисел
- для быстрой горизонтальной обработки символьной информации
- для автоматического распараллеливания вычислений

6. Программа, написанная на Delphi, выполняет сложение векторов целочисленных данных. Будет ли она выполняться быстрее на процессоре с поддержкой технологии MMX или SSE2?

- нет
- да
- только при условии поддержки расширений на уровне операционной системы
- только при указании параметров командной строки

7. Исполняемый файл программы оптимизирован под SSE. Будет ли она выполняться быстрее, если оказалось, что процессор также поддерживает технологии SSE2 или SSE3?

- нет
- да
- только при условии поддержки расширений на уровне операционной системы
- только при указании дополнительных параметров командной строки

8. Команды каких расширений могут выполняться непосредственно вперемешку с командами сопроцессора?

- SSE
- MMX
- 3DNow

9. При переходе с 32- на 64-битную платформу не изменяется

- количество MMX-регистров
- разрядность регистров общего назначения
- количество XMM-регистров
- объем адресуемой памяти

10. Расширение 3DNow предназначено

- для векторной обработки вещественных чисел с использованием MMX-регистров
- для векторной обработки целочисленных данных с использованием XMM-регистров
- для быстрой горизонтальной обработки символьной информации с использованием регистров общего назначения
- для автоматического распараллеливания вычислений

11. Команду EMMS необходимо использовать

- после команд MMX перед командами сопроцессора
- перед любыми командами сопроцессора
- после любых команд SIMD-расширений
- команда оставлена для обратной совместимости и ее использование не рекомендуется

12. Команду FWAIT рекомендуется использовать

- только в случае необходимости правильной обработки исключений
- после команд сопроцессора перед любыми другими командами
- для синхронизации процессора и сопроцессора
- для включения режима пониженного энергопотребления сопроцессора

13. Какая из перечисленных типов памяти видеокарт с поддержкой CUDA характеризуется максимальной скоростью работы?

- разделяемая
- глобальная
- оперативная
- текстурная
- дисковая

Шкала оценивания результатов тестирования: в соответствии с действующей в университете балльно-рейтинговой системой оценивание результатов промежуточной аттестации обучающихся осуществляется в рамках 100-балльной шкалы, при этом максимальный балл по промежуточной аттестации обучающихся по очной форме обучения составляет 36 баллов, по заочной форме обучения – 60 баллов (установлено положением П 02.016).

Максимальный балл за тестирование представляет собой разность двух чисел: максимального балла по промежуточной аттестации для данной формы обучения (36 или 60) и максимального балла за решение компетентностно-ориентированной задачи (6). Балл, полученный обучающимся за тестирование, суммируется с баллом, выставленным ему за решение компетентностно-ориентированной задачи. Общий балл по промежуточной аттестации суммируется с баллами, полученными обучающимся по результатам текущего контроля успеваемости в течение семестра; сумма баллов переводится в оценку по 5-балльной шкале следующим образом.

Соответствие 100-балльной и 5-балльной шкал

Сумма баллов по 100-балльной шкале	Оценка по 5-балльной шкале
100-85	отлично
84-70	хорошо
69-50	удовлетворительно
49 и менее	неудовлетворительно

2.2. Компетентностно-ориентированные задачи

1. Вывести на экран все простые числа в диапазоне от 1 до N . Число называется простым, если оно делится без остатка только само на себя и на 1.
2. Найти натуральное число в диапазоне от 1 до N с максимальной суммой делителей.
3. Представить заданное число в виде произведения простых чисел (например, $5040 = 2^4 \cdot 3^2 \cdot 5 \cdot 7$).
4. Найти наименьшее натуральное число N из диапазона $[A, B]$, представимое в виде суммы кубов двух натуральных чисел $N = x^3 + y^3$.
5. Натуральное число из N цифр называется числом Армстронга, если оно может быть представлено в виде суммы его цифр, возведенных в N -ую степень (например, $153 = 1^3 + 5^3 + 3^3$). Найти числа Армстронга в заданном диапазоне.
6. Указать все способы представления заданного числа N в виде суммы квадратов 3 натуральных чисел $N = x^2 + y^2 + z^2$.
7. Согласно теореме Лагранжа о сумме четырех квадратов любое число можно представить в виде суммы не более чем 4 квадратов натуральных чисел. Проверить справедливость теоремы для чисел из заданного диапазона.
8. (Плоские числа) Найти и вывести на экран все числа, меньшие N , единственным образом представимые в виде произведения простых чисел A и B (например, $985 = 5 \cdot 197$).
9. (Телесные числа) Найти и вывести на экран все числа, меньшие N , единственным образом представимые в виде произведения простых чисел A , B и C (например, $385 = 5 \cdot 7 \cdot 11$).

10. В приведенной последовательности сумм

$$S_1 = 1,$$

$$S_2 = 3 + 5 = 8,$$

$$S_3 = 7 + 9 + 11 = 27,$$

...

вычислить значение i -ой суммы. Вывести на экран все промежуточные значения.

11. Найти наилучшее приближение к заданному иррациональному числу (π , e , $\sqrt{2}$ и т.д.) в виде рациональной дроби $\frac{A}{B}$ в заданном диапазоне ($1 \leq A \leq N$, $1 \leq B \leq N$).

12. Проверить справедливость соотношения $\sum_{i=0}^n C_n^i = 2^n$, где $C_n^m = \frac{n!}{m!(n-m)!}$ – число сочетаний из n по m .

13. Проверить справедливость соотношения $\sum_{i=0}^n (-1)^i C_n^i = 0$, где $C_n^m = \frac{n!}{m!(n-m)!}$ – число сочетаний из n по m .

14. (Тождество Абеля) Убедиться в справедливости соотношения $2(n-1)n^{n-2} = \sum_{\substack{k+m=n, \\ k>1, m>1}} C_n^k k^{k-1} m^{m-1}$, где $C_n^m = \frac{n!}{m!(n-m)!}$ – число сочетаний из n по m .

15. Проверить справедливость соотношения $n! = 1 + \sum_{i=1}^{n-1} i \cdot i!$.

16. Проверить справедливость соотношения $\sum_{i=1}^n i^3 = \left(\sum_{i=1}^n i \right)^2 = \frac{n^2(n+1)^2}{4}$.

17. Проверить справедливость соотношения $\sum_{k=0}^{n-1} 2^k = 2^n - 1$.

18. Проверить справедливость соотношения $(x-a)^n \equiv x^n - a \pmod{n}$, где x, a, n – целые положительные числа, n – простое число, $a \leq x$, с использованием датчика случайных чисел.

19. (малая теорема Ферма) Если a не делится на простое число p , то $a^{p-1} - 1$ делится на p . Проверить справедливость теоремы с использованием датчика случайных чисел.

20. Проверить или опровергнуть гипотезу Ферма. Простые числа вида $4k+1$ (k – натуральное число) можно единственным способом представить в виде суммы квадратов двух целых чисел. Например, при $k=1$ $p=4 \cdot 1 + 1 = 5 = 1^2 + 2^2$, а при $k=3$ $p=4 \cdot 3 + 1 = 13 = 2^2 + 3^2$.

21. Найти все простые множители заданного числа N .

22. Для всех нечетных чисел n значение $n^2 - 1$ делится на 8. Проверить справедливость утверждения.

23. Если $p > 3$ – простое число, то значение $p^2 - 1$ делится на 24. Убедиться в справедливости утверждения.

24. Найти все возможные значения цифр $0 \leq D, E, M, N, O, R, S, Y \leq 9$ такие, что выполняется равенство

$$\begin{array}{r} SEND \\ + MORE \\ \hline MONEY \end{array}$$

Пример решения: $D = 0, E = 0, M = 1, N = 0, O = 0, R = 0, S = 9, Y = 0$.

25. Убедиться в справедливости равенства $\sum_{i=1}^n H_i = (n+1)[H_{n+1} - 1]$, где $H_n = \sum_{k=1}^n \frac{1}{k}$ – ряд гармонических чисел.
26. Убедиться в справедливости равенства $\sum_{i=1}^n iH_i = C_{n+1}^2 \left(H_{n+1} - \frac{1}{2} \right)$, где $C_n^m = \frac{n!}{m!(n-m)!}$ – число сочетаний из n по m , $H_n = \sum_{k=1}^n \frac{1}{k}$ – сумма ряда гармонических чисел.
27. Убедиться в справедливости равенства $\sum_{i=1}^n C_i^k H_i = C_{n+1}^{k+1} \left(H_{n+1} - \frac{1}{k+1} \right)$ для заданных значений n и k , где $C_n^m = \frac{n!}{m!(n-m)!}$ – число сочетаний из n по m , $H_n = \sum_{k=1}^n \frac{1}{k}$ – сумма ряда гармонических чисел.
28. Проверить справедливость соотношения $\sum_{k=0}^n x^k = \frac{x^{n+1} - 1}{x - 1}$.
29. Проверить справедливость соотношения $\sum_{k=1}^n \frac{1}{k(k+1)} = \frac{n}{n+1}$.
30. Проверить справедливость соотношения $\sum_{k=1}^n k \cdot 2^k = (n-1) \cdot 2^{n+1} + 2$.

Шкала оценивания решения компетентностно-ориентированной задачи: в соответствии с действующей в университете балльно-рейтинговой системой оценивание результатов промежуточной аттестации обучающихся осуществляется в рамках 100-балльной шкалы, при этом максимальный балл по промежуточной аттестации обучающихся по очной форме обучения составляет 36 баллов, по очно-заочной и заочной формам обучения – 60 (установлено положением П 02.016).

Максимальное количество баллов за решение компетентностно-ориентированной задачи – 6 баллов.

Балл, полученный обучающимся за решение компетентностно-ориентированной задачи, суммируется с баллом, выставленным ему по результатам тестирования.

Общий балл промежуточной аттестации суммируется с баллами, полученными обучающимся по результатам текущего контроля успеваемости в течение семестра; сумма баллов переводится в оценку по 5-балльной шкале.

Критерии оценивания решения компетентностно-ориентированной задачи:

6-5 баллов выставляется обучающемуся, если решение задачи демонстрирует глубокое понимание обучающимся предложенной проблемы и разностороннее ее рассмотрение; свободно конструируемая работа представляет собой логичное, ясное и при этом краткое, точное описание хода решения задачи и формулировку доказанного, правильного вывода (ответа); задача решена в установленное преподавателем время или с опережением времени.

4-3 балла выставляется обучающемуся, если решение задачи демонстрирует понимание обучающимся предложенной проблемы; задача решена типовым способом в установленное преподавателем время; имеют место несущественные недочеты в описании хода решения и (или) вывода (ответа).

2-1 балла выставляется обучающемуся, если решение задачи демонстрирует поверхностное понимание обучающимся предложенной проблемы; осуществлена попытка шаблонного решения задачи, но при ее решении допущены ошибки и (или) превышено установленное преподавателем время.

0 баллов выставляется обучающемуся, если решение задачи демонстрирует непонимание обучающимся предложенной проблемы и (или) задача не решена.

2.3. Банк вопросов для контроля выполнения лабораторных и/или практических работ

Грид-системы

1. В чем заключается проблема «последней мили»?
2. В чем отличия между настольными ЭВМ, вычислительными кластерами, суперкомпьютерами и грид-системами?
3. Чем отличаются сильно- и слабосвязанные задачи?
4. Какие современные актуальные вычислительно сложные задачи допускают эффективную параллельную реализацию с использованием грид-систем?
5. Что подразумевается под определением кворума при организации параллельных вычислений на платформе BOINC?
6. С какой целью выполняется валидация полученных результатов?

Методика измерения времени выполнения заданного фрагмента программы

1. С какой целью производится анализ времени выполнения заданных фрагментов программы?
2. Какие способы измерения времени выполнения существуют? В чем их отличия?
3. Для чего применяется изменение приоритета текущего потока или процесса?
4. Для чего применяется привязка потока к одному из ядер процессора?
5. Какие причины приводят к наличию погрешности в измеряемых величинах?

Введение в алгоритмическую и высокоуровневую оптимизацию

1. Что называется алгоритмической оптимизацией?
2. Какие оптимизации относятся к высокоуровневым?
3. Что является ключевым недостатком базового алгоритма умножения матриц?
4. За счет чего буферизованное умножение оказывается быстрее базового алгоритма?
5. Сколько операций с плавающей точкой необходимо выполнить при умножении двух плотных квадратных матриц размера ?

Введение в оптимизацию программных средств с использованием векторных расширений системы команд процессора

1. Что такое SIMD-принцип организации вычислительной системы?
2. Какие SIMD-расширения системы команд процессора существуют? В чем их ключевые отличия?
3. Какие регистры используются в составе SIMD-расширений системы команд процессора? Каковы их размер и логическая интерпретация?
4. За счет чего получается выигрыш во времени обработки при использовании SIMD-расширений?
5. Почему практический выигрыш во времени обработки ниже теоретического предела?

Разработка программ с поддержкой технологии CUDA с использованием компилятора командной строки

1. Для чего предназначен инструментарий CUDA?
2. Как производится программирование NVidia GPU с использованием CUDA?
3. Как производится сборка программы для ее запуска на NVidia GPU?
4. Что такое CUDA-ядро (CUDA kernel) и его конфигурация запуска?

Определение параметров видеокарты с поддержкой технологии CUDA в среде Microsoft Visual Studio

1. Для чего предназначен инструментарий CUDA?
2. Как производится программирование NVidia GPU с использованием CUDA?
3. Как производится сборка программы для ее запуска на NVidia GPU?
4. Как инструментарий CUDA интегрируется в состав среды Microsoft Visual Studio?

Измерение пропускной способности памяти видеокарт с поддержкой технологии CUDA

1. Для чего предназначен инструментарий CUDA?
2. Как производится программирование NVidia GPU с использованием CUDA?
3. Как производится обмен данными между процессором, оперативной памятью и видеокартой?
4. Какая подсистема в составе компьютера лимитирует скорость обмена данными между оперативной памятью и глобальной памятью видеокарты?

Оптимизация работы с глобальной памятью для видеокарт с поддержкой технологии CUDA

1. Для чего предназначен инструментарий CUDA?
2. Как производится программирование NVidia GPU с использованием CUDA?
3. В чем заключается особенность умножения матриц на GPU?
4. Чем лимитируется время выполнения умножения матриц для каждого из алгоритмов? Какой из способов умножения матриц является наиболее быстрым?
5. Какие типы памяти есть в составе GPU? В чем их ключевые особенности?

2.4. Вопросы к зачету

1. Краткая история развития параллельных вычислительных средств
2. Основные тенденции развития параллельных вычислительных средств
3. Производительность вычислительных средств и способы ее оценки
4. Профайлеры и профилирование программ. Способы измерения времени выполнения фрагментов кода. Поиск узких мест
5. Закон Амдала
6. Закон Густафсона-Барсиса
7. Способы повышения производительности аппаратных средств вычислительной техники и их программная поддержка
8. Виды параллельных вычислительных структур. Достоинства и недостатки. Особенности программирования
9. Оптимизация программных средств: уровни оптимизации и основные правила оптимизации
10. Оптимизирующие компиляторы: обзор возможностей, поддерживаемые оптимизации
11. Грид-системы: принципы организации и программирования. Добровольный метакомпьютинг
12. Векторные расширения системы команд процессоров семейства x86
13. Программирование с использованием расширения MMX
14. Программирование с использованием расширения 3DNow
15. Программирование с использованием расширения SSE
16. Программирование с использованием расширений SSE2, SSE3, SSSE3
17. Программирование многоядерных и многопроцессорных систем. Понятие потоков и процессов. Примитивы синхронизации и межпроцессного обмена
18. Многопоточное программирование. Понятие потоков и процессов. Приоритеты, привязка к ядрам процессора. Примитивы синхронизации

19. Программирование с использованием интерфейса MPI. Обмены данными, виды обменов, топологии.
20. Программирование в рамках концепции GPGPU. Обмен данными с оперативной памятью, виды памяти GPU, организация параллельного исполнения кода. Синхронизация нитей.

Критерии оценки:

- ✓ «0» баллов выставляется обучающемуся за отсутствие ответов или отказ от ответа.
- ✓ «2» балла выставляется за фрагментарные невязанные знания по предмету, обрывочный пересказ с низкой степенью осмысления, отсутствие ответов на наводящие вопросы преподавателя, некомпетентность в установленной терминологии и обозначениях.
- ✓ «4» балл выставляется обучающемуся за неумение ориентироваться в основных положениях дисциплины, использование установленной терминологии с существенными стилистическими и логическими ошибками. содержание материала излагается поверхностно, неполно, без логической последовательности, несамостоятельно, в ответах на вопросы присутствуют существенные логические ошибки.
- ✓ «6» балла выставляется за достаточный объем знаний в рамках дисциплины, использование установленной терминологии и изложение ответов на вопросы без грубых ошибок, воспроизведение фактического и теоретического материала без обобщений и выводов, умение ориентироваться в основных концепциях и понятиях дисциплины, приводить типовые примеры.
- ✓ «8» балла – воспроизведение фактического и теоретического учебного материала последовательное, точное, осмысленное, не совсем самостоятельное, с несущественными ошибками и неточностями, способность самостоятельно приводить поясняющие примеры, владение инструментарием дисциплины, умение давать краткую сравнительную оценку и общие выводы, умение устанавливать причинно-следственные связи при анализе конструкторских и технологических решений.
- ✓ «12» балла – достаточно полные знания по дисциплине, содержание материала излагается последовательно, точно, правильно, осмысленно, самостоятельно, грамотное использование необходимой научной терминологии, умение делать обоснованные выводы, способность выявлять главенствующие факторы при техническом анализе вопросов. Даются ответы на любые заданные вопросы с несущественными ошибками и недочетами.
- ✓ «14» балла – изложение материала сжатое, структурированное в соответствии с собственной логической схемой учащегося, владение программным материалом высокой степени сложности и оперирование им в знакомой ситуации, владение инструментарием дисциплины для постановки и решения научных и профессиональных задач в различных условиях производства.
- ✓ «16» баллов выставляется за системное, образное, доказательное изложение материала, с использованием собственных схем и материала, выходящего за пределы вопросов курса, безупречное владение инструментарием дисциплины, умение оперативно и осознанно трансформировать полученные знания для решения проблем и задач в нестандартной ситуации. Владение системным подходом к анализу технических методов и пр.