Билет №78

**1. В каких величинах проводится оценки производительности** **многопроцессорных вычислительных систем?**

Количество вычислительных операций с плавающей запятой в секунду.

Количество выполняемых простых синтетических задач в секунду.

Количество регистровых операций в секунду.

**2. Критериями оценки эффективности многопроцессорных вычислительных систем являются:**

Масштабируемость, Отказоустойчивость, Производительность.

Многопроцессорность, Надежность, Вычислительная мощность.

Надежность, Производительность, Мультипроцессность.

**5. Масштабируемость вычислительных систем – это**

возможность пропорционального увеличения общей производительности системы путем добавления соответствующих аппаратных ресурсов;

масштабное увеличение мощности за определенный промежуток времени (закон Мура);

разбиение сложной задачи на множество подзадач.

**7. Разделяемую общую память с единым адресным пространством имеют**

кластерные системы

все процессоры SMP

массово-параллельные системы

SIMD-архитектуры

**1. Какие принципы технической реализации вычислительных систем (ВС) относятся к базовым?**

Модульность, Близкодействие.

Масштабируемость, Децентрализованность.

Асинхронность, децентрализованность, распределенность.

**2. Каким требованиям удовлетворяют вычислительные системы, основанные на принципах модульности и близкодействия?**

Асинхронности, Распределенности.

Синхронности, Распределенности.

Локальности, Синхронности.

**7. Чем достигается децентрализованность управления ВС?**

Отсутствием выделенного модуля как единого для всей системы центра управления.

Совместной работой всех исправных модулей системы.

Способностью каждого модуля принимать решения независимо от других модулей.

**8. Что понимается под масштабируемостью (Scalability) вычислительных систем?**

Способность к наращиванию и сокращению ресурсов.

Масштабированное изменение количества процессоров в ВС по арифметической или геометрической прогрессии.

Способность к наращиванию и сокращению вычислительной мощности ВС за счёт масштабирования точности вычислений.

**6. Как определяется понятие ускорения?**

Отношение времени выполнения последовательного алгоритма к времени выполнения параллельного алгоритма.

Отношение времени выполнения последовательного алгоритма к количеству используемых процессоров.

Произведение времени выполнения параллельного алгоритма к количеству используемых процессоров.

**7. Как определяется понятие эффективности?**

Отношение времени выполнения параллельного алгоритма к количеству используемых процессоров.

Отношение времени выполнения последовательного алгоритма к времени выполнения параллельного алгоритма.

Отношение времени выполнения стоимостно-оптимального параллельного алгоритма к ускорению.

**8. Возможно ли достижения сверхлинейного ускорения?**

Да

Нет

**12. В чем состоит понятие стоимостно-оптимального алгоритма?**

Алгоритм, стоимость которого является пропорциональной времени выполнения наилучшего последовательного алгоритма.

Алгоритм, стоимость которого является пропорциональной времени выполнения наилучшего параллельного алгоритма.

Алгоритм, стоимость которого является пропорциональной количеству процессоров.

**1. Существенные показатели для вычислительных кластеров**

a) Высокая производительность процессора при выполнении операций над числами с плавающей точкой.

b) Высокая латентность объединяющей сети.

c)

**4. Для какой топологии сети передачи данных при p-процессорах диаметр сети равен [p/2]**

Кольцо

Звезда

Линейка

**7. При передаче данных от всех процессоров всем процессорам сети (all-to-all broadcast) общая длительность операции рассылки сообщений в топологии решетка-тор:**

*tпд* = *2(tн + mtk) ( √p /2).*

*tпд* = *2tн ( p −1) + mtk ( p −1).*

*tпд* = (*tн + mtk )( p /2).*

**8. Особенности метода передачи пакетов по сравнению с методом передачи сообщений:**

Уменьшает потребность в памяти для хранения пересылаемых данных.

Увеличивает время пересылки данных из-за увеличения накладных расходов (время подготовки и время передачи служебных данных на каждый пакет).

Потребует разработки менее сложного аппаратного и программного обеспечения сети.

**1. Выберите правильную последовательность организации разработки параллельных алгоритмов**

Разделение вычислений на независимые части, выделение информационных зависимостей, масштабирование подзадач.

Распределение подзадач между процессорами, выделение информационных зависимостей, разделение вычислений на независимые части.

Выполнить анализ имеющихся вычислительных схем, Распределение подзадач между процессорами, выделение информационных зависимостей.

**3. Увеличивает ли объём общий вычислений и требуемый объём памяти** **выполненная декомпозиция?**

Не увеличивает.

Увеличивает и общий объём вычислений, и требуемый объём памяти

Увеличивает только общий объём вычислений.

**4. Возможна ли при выбранном способе декомпозиции равномерная загрузка всех имеющихся процессоров?**

Возможна, если в схеме вычислений отсутствуют последовательные участки.

Возможна, при использовании крупноблочного распараллеливания.

Равномерная загрузка всех имеющихся процессоров не возможна.

**8. При выполнении декомпозиции разделение данных происходит по:**

ленточной схеме;

кольцевой схеме;

блочной схеме.

**4. Что понимается под понятием потока (thread)?**

Наименьшая единица обработки, исполнение которой может быть назначено ядром операционной системы.

Экземпляр программы во время выполнения, независимый объект, которому выделены системные ресурсы (например, процессорное время и память).

**8. В чем состоят понятия фрагмента параллельной программы?**

Блок программы, управляемый директивой parallel.

Параллельно выполняемые участки программного кода, динамически-возникающие в результате вызова функций.

Часть параллельного блока программы, выделяемая для параллельного выполнения.

**10. В чем состоят понятия области параллельной программы?**

Параллельно выполняемые участки программного кода, динамически-возникающие в результате вызова функций из параллельных фрагментов.

Параллельные фрагменты, представляющие параллельно-выполняемую часть программы.

Часть параллельного фрагмента, выделяемая для параллельного выполнения при помощи директивы parallel.

**11. В чем состоят понятия секции параллельной программы?**

Часть параллельного фрагмента, выделяемая для параллельного выполнения при помощи директивы section.

Блок программы, управляемый директивой section.

Параллельно выполняемые участки программного кода, динамически-возникающие в результате вызова функций section.

**4. Как определяется порядок выполнения итераций в распараллеливаемых циклах в OpenMP?**

При помощи параметра schedule директивы for.

При помощи параметра distribute директивы for.

При помощи параметра shared директивы for.

**6. В OpenMP синхронизация может быть обеспечена при помощи**

директивы **barrier.**

специального типа семафоров – замков (**locks**).

директивы **private**.

**7. Что понимается под операцией редукции?**

Совместная обработка локальных переменных.

Сохранение значения локальной переменной одного из потоков.

Задание набора локальных переменных.

**8. Как определяется критическая секция?**

a) Блок программного кода, который может выполняться только одним потоком в каждый конкретный момент времени.

b) Блок программного кода, где выполняется **с**овместная обработка локальных переменных.

c) Фрагмент программы, где изменение значений общих переменных должно осуществляться в каждый конкретный момент времени только одним потоком.

**2. Что следует понимать под параллельной программой?**

Множество одновременно выполняемых процессов.

Параллельный алгоритм задачи.

Программа, реализованная на многопроцессорной системе.

**6. Какая функция MPI обеспечивает передачу данных от одного процесса всем процессам?**

MPI\_Bcast.

MPI\_Send\_ALL.

MPI\_ASend.

**10. Что понимается в MPI под коммуникатором?**

Служебный объект, который объединяет в своем составе группу процессови их контекст.

Программно-управляемое устройство выполняющее функцию маршрутизации.

Программа, управляющая режимами обмена (парные, коллективные).

**11. Что понимается в MPI под виртуальной топологией?**

Структура линий связи между процессамив виде полного графа.

Геометрическая форма плоской проекции среды передачи.

Запрограммированная топология обменов в соответствии с информационной структурой задачи (независимо от наличия реальных физических каналов связи между процессорами)