Государственное образовательное учреждение
«Приднестровский государственный университет им. Т.Г.Шевченко»
Инженерно-технический институт

Кафедра «Информационные технологии и автоматизированное управление производственными процессами»

УТВЕРЖДАЮ

Заведующий кафедрой ИТиАУПП

Ю.А. Столяренко

«28» августа 2020 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по дисциплине

Б1.О.09 «ПАРАЛЛЕЛЬНЫЕ МЕТОДЫ И АЛГОРИТМЫ»

Направление подготовки
2.09.04.01 «Информатика и вычислительная техника»

Профиль (специализация) подготовки
Информационное и программное обеспечение вычислительных систем

Квалификация (степень) выпускника: **магистр**

Форма обучения: **очная, заочная**

Год набора: **2020 г.**

Разработал: доцент

А.Ю. Долгов

«28» августа 2020 г.

Тирасполь, 2020

Паспорт фонда оценочных средств по учебной дисциплине.

1. В результате изучения «Параллельные методы и алгоритмы» у обучающихся должны быть сформированы следующие компетенции:

Категория (группа) компетенций	Код и наименование	Код и наименование индикатора достижения универсальной компетенции
Общепрофессиональные компетенции и индикаторы их достижения		
	ОПК-2 Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий для решения профессиональных задач	ИД-1$_{ОПК-2}$ Знать современные интеллектуальные технологии для решения профессиональных задач;
		ИД-2$_{ОПК-2}$ Уметь обосновать выбор современных интеллектуальных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач;
		ИД-3$_{ОПК-2}$ Иметь навыки разработки оригинальных программных средств, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач.
	ОПК-5. Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем	ИД-1$_{ОПК-5}$ Знать современное программное и аппаратное обеспечение информационных и автоматизированных систем
		ИД-2$_{ОПК-5}$ Уметь модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач;
		ИД-3$_{ОПК-5}$ Иметь навыки разработки программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач
	ОПК-6. Способен разрабатывать компоненты программно-аппаратных комплексов обработки информации и автоматизированного проектирования	ИД-1$_{ОПК-6}$ Знать: аппаратные средства и платформы инфраструктуры информационных технологий, виды, назначение, архитектуру, методы разработки и администрирования программно-аппаратных комплексов объекта профессиональной деятельности
		ИД-2$_{ОПК-6}$ Уметь: анализировать техническое задание, разрабатывать и оптимизировать программный код для решения задач обработки информации и автоматизированного проектирования
		ИД-3$_{ОПК-6}$ Владеть: навыками составления технической документации по использованию и настройке компонентов программно-аппаратного комплекса

2. Программа оценивания контролируемой компетенции:

Текущая аттестация	Контролируемые модули, разделы (темы) дисциплины и их наименования	Код контролируемой компетенции или его части	Наименование оценочного средства
РУБЕЖНЫЙ КОНТРОЛЬ	Архитектура параллельных вычислительных систем	ОПК-2, ОПК-5	Т1, ЛР1

	Программное обеспечение параллельных вычислительных систем	ОПК-5,ОПК-6	Т1, Т2, ЛР2, ЛР3
РУБЕЖНАЯ АТТЕСТАЦИЯ	Параллельные методы и алгоритмы	ОПК-2,ОПК-6	Т2, ЛР4
Промежуточная аттестация		Код контролируемой компетенции или его части	Наименование оценочного средства
		ОПК-2, ОПК-5, ОПК-6	зачет

3. Показатели и критерии оценивания компетенции по этапам формирования, описание шкал оценивания

Этапы оценивания компетенции	Показатели достижения заданного уровня освоения компетенции	Критерии оценивания результатов обучения			
		2	**3**	**4**	**5**
Первый этап	ИД-1$_{ОПК-2}$ **Знать** современные интеллектуальные технологии для решения профессиональных задач	Не знает	Знает основные понятия теории, но не знает способы применения современных интеллектуальных технологий для решения профессиональных задач	Знает основные понятия и основы теории, но не может применять одну из современных интеллектуальных технологий для решения профессиональных задач	Знает основные понятия и основы теории. Умеет применять современные интеллектуальные технологии для решения профессиональных задач
Второй этап	ИД-2$_{ОПК-2}$ **Уметь** обосновать выбор современных интеллектуальных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач	Не умеет	Правильно определяет выбор современных интеллектуальных технологий, но не умеет обосновать выбор программной среды при разработке оригинальных программных средств для решения профессиональных задач	Умеет обосновать выбор современных интеллектуальных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач, но не умеет обрабатывать результаты	Умеет обосновать выбор современных интеллектуальных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач
Третий этап	ИД-3$_{ОПК-2}$ **Владеть** навыки разработки оригинальных программных средств, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	Не владеет	Владеет частью навыков разработки оригинальных программных средств, но не владеет навыками использования современных интеллектуальных технологий, для решения профессиональных задач	Владеет частью навыков разработки оригинальных программных средств, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	Владеет навыки разработки оригинальных программных средств, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач

Первый этап	ИД-1$_{ОПК-5}$ **Знать** современное программное и аппаратное обеспечение информационных и автоматизированных систем	Не знает	Знает основные понятия теории, но не знает способы применения современного программного и аппаратного обеспечения информационных и автоматизированных систем	Знает основные понятия и основы теории современного программного и аппаратного обеспечения информационных и автоматизированных систем	Знает современное программное и аппаратное обеспечение информационных и автоматизированных систем
Второй этап	ИД-2$_{ОПК-5}$ **Уметь** модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач	Не умеет	Правильно анализирует возможности модернизации программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач, но не умеет модернизировать программный код и структуру аппаратного обеспечения	Умеет модернизировать либо программное, либо аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач	Умеет модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач
Третий этап	ИД-3$_{ОПК-5}$ **Владеть** навыками разработки программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач	Не владеет	Владеет частью навыков разработки либо программного, либо аппаратного обеспечения информационных и автоматизированных систем, но не владеет навыками их применения для решения профессиональных задач	Владеет навыками составления разработки либо программного, либо аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач	Владеет навыками разработки программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач
Первый этап	ИД-1$_{ОПК-6}$ **Знать:** аппаратные средства и платформы инфраструктуры информационных технологий, виды, назначение, архитектуру, методы разработки и администрирования программно-аппаратных	Не знает	Знает основные понятия теории, но не знает способы применения аппаратных средств и платформ инфраструктуры информационных технологий, виды, назначение, архитектуру, методы разработки и	Знает основные понятия и основы теории аппаратных средств и платформ инфраструктуры информационных технологий, виды, назначение, архитектуру, методы разработки и администрирования программно-аппаратных комплексов объекта	Знает аппаратные средства и платформы инфраструктуры информационных технологий, виды, назначение, архитектуру, методы разработки и администрирования программно-аппаратных комплексов объекта профессиональной деятельности

	комплексов объекта профессиональной деятельности		администрирования программно-аппаратных комплексов объекта профессиональной деятельности	профессиональной деятельности	
Второй этап	ИД-2$_{ОПК-6}$ **Уметь:** анализировать техническое задание, разрабатывать и оптимизировать программный код для решения задач обработки информации и автоматизированного проектирования	Не умеет	Правильно анализирует техническое задание, но не умеет разрабатывать и оптимизировать программный код для решения задач обработки информации и автоматизированного проектирования	Умеет анализировать техническое задание и разрабатывать программный код для решения задач обработки информации и автоматизированного проектирования, но не умеет их оптимизировать.	Умеет анализировать техническое задание, разрабатывать и оптимизировать программный код для решения задач обработки информации и автоматизированного проектирования
Третий этап	ИД-3$_{ОПК-6}$ **Владеть:** навыками составления технической документации по использованию и настройке компонентов программно-аппаратного комплекса	Не владеет	Владеет навыками составления технической документации по использованию компонентов программно-аппаратного комплекса, но не владеет навыками их настройки	Владеет навыками составления технической документации по использованию и настройке компонентов либо программной, либо аппаратной составляющей комплекса	Владеет навыками составления технической документации по использованию и настройке компонентов программно-аппаратного комплекса

4. Шкала оценивания

Согласно Положению «О порядке организации аттестации в ИТИ ПГУ им. Т.Г. Шевченко, итоговая оценка представляет собой сумму баллов, полученных студентом по итогу освоения дисциплины (модуля):

Оценка в традиционной шкале	Оценка в 100-балльной шкале	Буквенные эквиваленты оценок в шкале ЗЕ (% успешно аттестованных)
5 (отлично)	88–100	A (отлично) – 88-100 баллов
4 (хорошо)	70–87	B (очень хорошо) – 80-87баллов
		C (хорошо) – 70-79 баллов
3 (удовлетворительно)	50–69	D(удовлетворительно) – 60-69 баллов
		E(посредственно) – 50-59 баллов
2 (неудовлетворительно)	0–49	Fx– неудовлетворительно, с возможной пересдачей – 21-49 баллов
		F– неудовлетворительно, с повторным изучением дисциплины – 0-20 баллов

Расшифровка уровня знаний, соответствующего полученным баллам, дается в таблице, указанной ниже

A	“Отлично” - 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.
B	“Очень хорошо” - теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом в основном сформированы, все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному.
C	“Хорошо” - 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 виды заданий выполнены с ошибками.
D	“Удовлетворительно” - 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 из выполненных заданий, возможно, содержат ошибки.
E	“Посредственно” - теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному.
FX	“Условно неудовлетворительно” - теоретическое содержание курса освоено частично, необходимые практические навыки работы не сформированы, большинство предусмотренных программой обучения учебных заданий не выполнено, либо качество их выполнения оценено числом баллов, близким к минимальному; при дополнительной самостоятельной работе над материалом курса возможно повышение качества выполнения учебных заданий.
F	“Безусловно неудовлетворительно” - теоретическое содержание курса не освоено, необходимые практические навыки работы не сформированы, все выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к какому-либо значимому повышению качества выполнения учебных заданий.

5. **Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций при изучении учебной дисциплины в процессе освоения образовательной программы**

5.1 Типовой вариант задания на контрольную работу

5.2. Типовой вариант задания на практическую работу

5.3 Типовой тест промежуточной аттестации

Перечислите классификационные признаки, используемые в классификации вычислительных систем Флинна.

(Найдите соответствие)

Одиночный поток команд	В один момент времени может выполняться только одна команда. В этом случае эта единственная команда определяет в данный момент времени работу всех или, по крайней мере, многих устройств вычислительной системы
Множественный поток команд	В один момент времени может выполняться много команд. В этом случае каждая из таких команд определяет в данный момент времени работу только одного или лишь нескольких (но не всех)

	устройств вычислительной системы.
Одиночный поток данных	Предполагает наличие в вычислительной системе только одного устройства оперативной памяти и одного процессора. Однако при этом процессор может быть как угодно сложным, так что процесс обработки каждой единицы информации в потоке может требовать выполнения многих команд.
Множественный поток данных	Состоит из многих зависимых или независимых одиночных потоков данных.

2. Перечислите типы вычислительных систем по классификации Флинна.

(Внимание! Множественный ответ)

+1) SISD-система представляет собой классическую однопроцессорную ЭВМ фон неймановской архитектуры.

2) SIHD-системы – это системы с распределенной памятью, которые называются также слабосвязанными системами.

+3) MISD-система – векторно-конвейерные вычислительные система.

+4) SIMD-система содержит много процессоров, которые синхронно (как правило) выполняют одну и ту же команду над разными данными. SIMD-системы делятся на два больших класса: векторно-конвейерные вычислительные системы; векторно-параллельные вычислительные системы.

5) MIHD-системы – это системы с общей памятью, которые называются мультипроцессорными вычислительными системами или мультипроцессорами.

+6) MIMD-система содержит много процессоров, которые (как правило, асинхронно) выполняют разные команды над разными данными. Подавляющее большинство современных суперЭВМ имеют архитектуру MIMD (по крайней мере, на верхнем уровне иерархии). MIMD-системы часто называют многопроцессорными системами.

3. Назовите три основных класса параллельных вычислительных систем в зависимости от типа строения их оперативной памяти.

(Внимание! множественный ответ)

1) Вычислительные системы с общим процессорным пространством (Common Processor Area) являются наиболее значимыми в процессе строительства многопроцессорных систем. Они дают возможность объединять с помощью некоторой коммуникационной сети множество процессоров, которые в каждом такте синхронно выполняют одну и ту же команду над разными данными. Вычислительные системы с общим процессорным пространством имеют в своих системах команд специальные векторные (матричные) операции, такие, как векторное и матричное сложение, умножение вектора на матрицу, умножение матрицы на константу, вычисление скалярного произведения, свертки и т.д. При выполнении вычислительных операций различные компоненты векторов и матриц обрабатываются параллельно на различных процессорных элементах.

+2) В вычислительных системах с общей памятью (Common Memory Systems или Shared Memory Systems) значение, записанное в память одним из процессоров, напрямую доступно для другого процессора. Общая память обычно имеет высокую пропускную способность памяти (bandwidth) и низкую латентность памяти (latency) при передаче информации между процессорами, но при условии, что не происходит одновременного обращения нескольких процессоров к одному и тому же элементу памяти. К общей памяти доступ разных процессорами системы осуществляется, как правило, за одинаковое время. Поэтому такая память называется еще UMA–память (Unified Memory Access) – памятью с одинаковым временем доступа. Системы с такой памятью называются UMA-системами. Системы с общей памятью называются также сильносвязанными системами. Вычислительные системы с общей памятью называются мультипроцессорными вычислительными системами или мультипроцессорами.

+3) В системах с распределенной памятью (Distributed Memory Systems) каждый процессор имеет свою локальную память с локальным адресным пространством. Для систем с распределенной памятью характерно наличие большого числа быстрых каналов, которые связывают отдельные части этой памяти с отдельными процессорами. Обмен информацией между частями распределенной памяти осуществляется обычно относительно медленно. Системы с распределенной памятью называются также слабосвязанными системами. Вычислительные системы с распределенной памятью называются мультикомпьютерными вычислительными системами или мультикомпьютерами.

4) Вычислительные системы с общей коммуникационной сетью (Common Network Area) представляют собой системы, которые образуют единую транспортную систему передачи адресов, команд и данных. В этом процессе важное место занимает частота занятости и освобождения конкретного процессора при выполнении управляющей команды. Связывает между собой непосредственно или через сетевые коммутаторы сетевые адаптеры процессорных узлов, что дает возможность проектирования управляющих команд и специального программирования вычислительных процедур.

+5) Системы с гибридной памятью – NUMA-системы (Non-Uniform Memory Access Systems) имеют память, которая физически распределена по различным частям системы, но логически разделяема (образует единое адресное пространство). Такая память называется еще логически общей (разделяемой) памятью (logically shared memory). В отличие от UMA-систем, в NUMA-системах время доступа к различным частям оперативной памяти различно.

4. Назовите основные классы SIMD-вычислительных систем.

(Внимание! Множественный ответ)

+1) Основными принципами, заложенными в архитектуру векторно-конвейерных систем, являются конвейерная организация обработки потока команд и введение в систему команд набора векторных операций, которые позволяют оперировать с целыми массивами данных. Основу векторно-конвейерных систем составляют конвейеры операций (арифметических и иных), например, конвейеры сложения, умножения и деления вещественных чисел.

+2) Основу векторно-параллельных систем составляют векторно-параллельные процессоры, представляющие собой совокупность скалярных процессоров (процессорных элементов), которые объединены некоторой коммуникационной сетью и в каждом такте синхронно выполняют одну и ту же команду над разными данными. Векторно-параллельные процессоры имеют в своих системах команд специальные векторные (матричные) операции, такие, как векторное и матричное сложение, умножение вектора на матрицу, умножение матрицы на константу, вычисление скалярного произведения, свертки и т.д. При выполнении векторных операций различные компоненты векторов и матриц обрабатываются параллельно на различных процессорных элементах.

3) Основными классами SIMD-вычислительных систем на мультикомпьютерах являются: 1)однородные мультикомпьютеры с распределенной памятью - вычислительные системы с массивно-параллельной архитектурой (MPP-системы); 2)кластерные системы (вычислительные кластеры), которые состоят из совокупности компьютеров (персональных компьютеров или рабочих станций), объединенных локальной сетью в качестве коммуникационной среды

5. Перечислите основные классы MIMD-вычислительных систем.

(Внимание! Множественный ответ)

+1) Основными классами MIMD-вычислительных систем на мультипроцессорах являются: 1) однородные мультипроцессоры с равноправным (симметричным) доступом к общей оперативной памяти, которые называются SMP-системами; 2) NUMA-системы, в которых обеспечен логически общий доступ к данным, находящимся в физически распределенной памяти.

+2) Основными классами MIMD-вычислительных систем на мультикомпьютерах являются: 1)однородные мультикомпьютеры с распределенной памятью - вычислительные системы с массивно-параллельной архитектурой (MPP-системы); 2)кластерные системы (вычислительные кластеры), которые состоят из совокупности компьютеров (персональных компьютеров или рабочих станций), объединенных локальной сетью в качестве коммуникационной среды.

3) Основными принципами MIMD-вычислительных систем, заложенными в архитектуру векторно-конвейерных систем, являются конвейерная организация обработки потока команд и введение в систему команд набора векторных операций, которые позволяют оперировать с целыми массивами данных. Основу векторно-конвейерных систем составляют конвейеры операций (арифметических и иных), например, конвейеры сложения, умножения и деления вещественных чисел.

6. Назовите основные классификационные признаки, используемые при классификации вычислительных кластеров

При классификации вычислительных кластеров используются следующие основные классификационные признаки:

(Внимание! множественный ответ)

1) Основными принципами, заложенными в архитектуру векторно-конвейерных систем, являются конвейерная организация обработки потока команд и введение в систему команд набора векторных операций, которые позволяют оперировать с целыми массивами данных. Основу векторно-конвейерных систем составляют конвейеры операций (арифметических и иных), например, конвейеры сложения, умножения и деления вещественных чисел.

+2) Тип узловых процессоров. В качестве узлов вычислительного кластера могут использоваться персональные компьютеры, рабочие станции и SMP сервера. Если в качестве узлов кластера используются SMP системы, то такой вычислительный кластер называется SMP-кластером.

+3) Однородность узловых процессоров. Как и всякие MIMD-системы, вычислительные кластеры разделяются на однородные кластерные системы (однородные вычислительные кластеры) и гетерогенные кластерные системы (гетерогенные вычислительные кластеры). Обычно, когда говорят о вычислительных кластерах, подразумевают однородные вычислительные кластеры.

4) Производительность при равенстве нулю коммуникационных затрат. Асимптотическое быстродействие векторно-конвейерных систем и векторно-параллельных систем "достигается" в случае, когда обрабатываемые векторы имеют бесконечную длину. Если векторно-параллельная система или MIMD-система имеют N процессоров, то асимптотическое быстродействие таких систем в N раз выше производительности их одного процессора

+5) Функциональная направленность кластеров. С функциональной точки зрения кластерные системы разделяют на высокоскоростные кластерные системы (HP-кластеры) и кластерные системы высокой готовности – HA-кластеры.

7. Определите асимптотическое быстродействие в качестве основной меры производительности параллельных вычислительных систем.

+1) Асимптотическое быстродействие параллельных вычислительных систем – это производительность при равенстве нулю коммуникационных затрат. Асимптотическое быстродействие векторно-конвейерных систем и векторно-параллельных систем "достигается" в случае, когда обрабатываемые векторы имеют бесконечную длину. Если векторно-параллельная система или MIMD-система имеют N процессоров, то асимптотическое быстродействие таких систем в N раз выше производительности их одного процессора.

2) Асимптотическое быстродействие параллельных вычислительных систем – это длина векторов, на которых достигается половина асимптотического быстродействия параллельной вычислительной системы. Асимптотическое быстродействие, главным образом, характеризует технологию изготовления ЭВМ, в то время как длина полупроизводительности представляет собой критерий степени параллелизма системы.

3) Асимптотическое быстродействие параллельных вычислительных систем – это конвейерная организация обработки потока команд и введение в систему команд набора векторных операций, которые позволяют оперировать с целыми массивами данных. Основу векторно-конвейерных систем составляют конвейеры операций (арифметических и иных), например, конвейеры сложения, умножения и деления вещественных чисел.

8. Сформулируйте гипотезу Минского и назовите основные причины невыполнения ее в современных параллельных вычислительных системах.

1) Гипотеза Минского формулируется следующим образом. В N-процессорной векторно-конвейерной системе или SIMD-вычислительной системе, в которой производительность каждого процессора больше единицы, общая производительность системы растет как $2log_2N+1$.

В современных системах имеет место явление "обогащения" – начиная со трех процессоров производительность системы растет, а после пятого процессора – падает. Основными причинами невыполнения гипотезы Минского являются: рост количества процессоров не дает расти коммуникационным расходам; с ростом количества процессоров растет сбалансированность их загрузки.

+2) Гипотеза Минского формулируется следующим образом. В N-процессорной векторно-параллельной системе или MIMD-вычислительной системе, в которой производительность каждого процессора равна единице, общая производительность системы растет как log_2N.

В современных системах с большим количеством процессоров имеет место явление "насыщения" – начиная с некоторого количества процессоров производительность системы падает. Основными причинами невыполнения гипотезы Минского являются: рост количества процессоров приводит к росту коммуникационных расходов (вследствие роста диаметра коммуникационной сети); с ростом количества процессоров растет несбалансированность их загрузки.

3) Гипотеза Минского формулируется следующим образом. В *N*-процессорной процессорно-параллельной системе или SIMD-вычислительной системе, в которой производительность каждого процессора меньше единицы, общая производительность системы растет как $3log_2N$-1.

В многопоточных системах с умеренным количеством процессоров имеет место явление "недостатка" – начиная с некоторого количества процессоров производительность системы растет. Основными причинами невыполнения гипотезы Минского являются: рост количества процессоров приводит к росту коммуникационных расходов (изменение диаметра коммуникационной сети); с ростом количества узлов растет стабильность их загрузки за счет перераспределения.

9. Перечислите основные компоненты коммуникационной среды параллельной вычислительной системы, а также их основные функции.

(Найдите соответствие)

Основными компонентами коммуникационной среды параллельной вычислительной системы являются:

Сетевые адаптеры	Одной из основных функций данных устройств является обеспечение трансляционных обменов данными в сети.
Коммуникационная сеть	Связывает между собой непосредственно или через сетевые коммутаторы сетевые адаптеры процессорных узлов.
Сетевые коммутаторы	Предназначены для приема данных от одного из узлов коммуникационной сети и передаче их некоторому другому узлу этой сети.

10. Дайте определения расстояния между процессорами в коммуникационной сети и диаметра и коммуникационной сети.

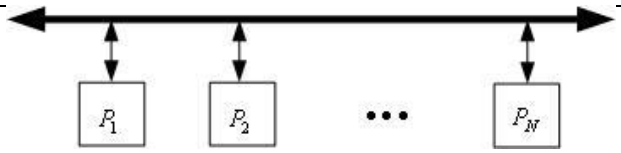
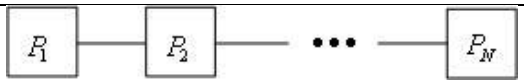
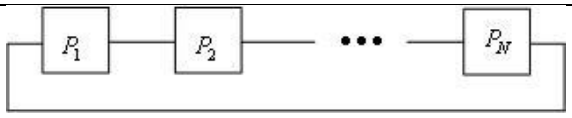
+1) Расстояние между узлами коммуникационной сети – количество ребер, которые образуют кратчайший путь между двумя узлами в графе коммуникационной сети. Диаметр коммуникационной сети – это максимальное расстояние между двумя произвольными узлами этой сети.

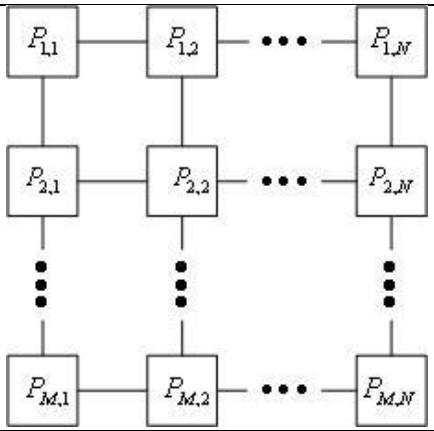
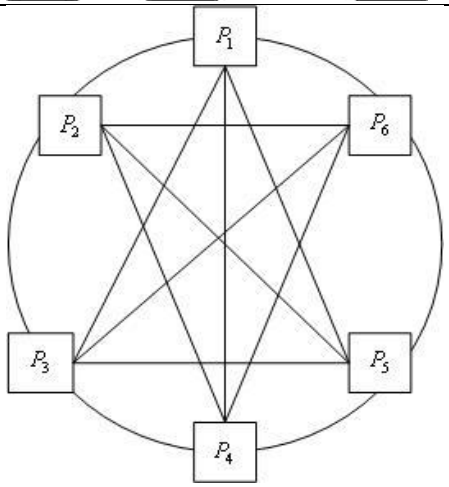
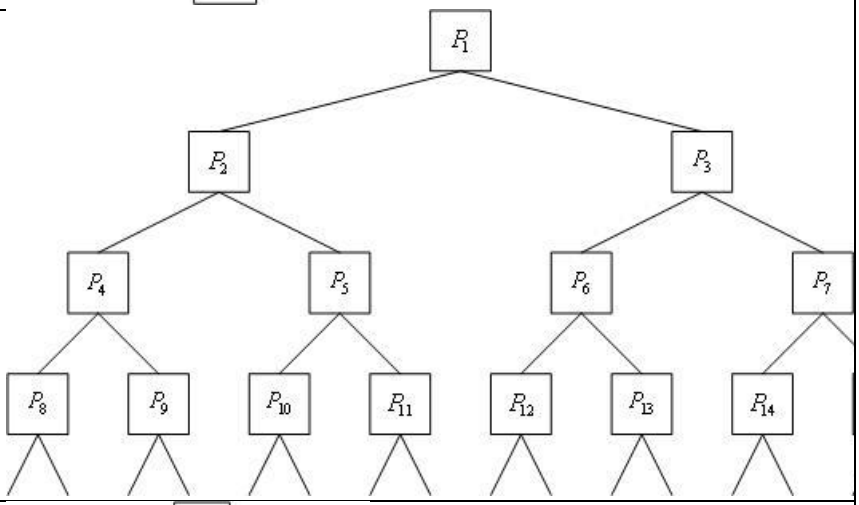
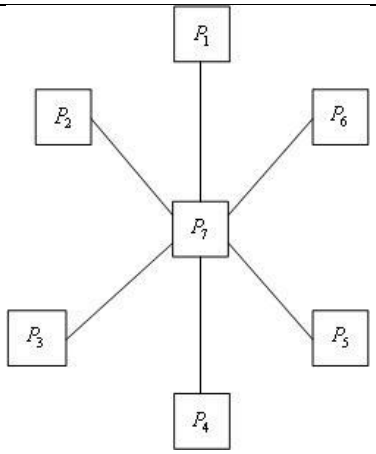
2) Расстояние между узлами коммуникационной сети – количество ядер, которые образуют группировку из двух и более узлов в графе коммуникационной сети. Диаметр коммуникационной сети – это минимальное расстояние между двумя произвольными узлами этой сети.

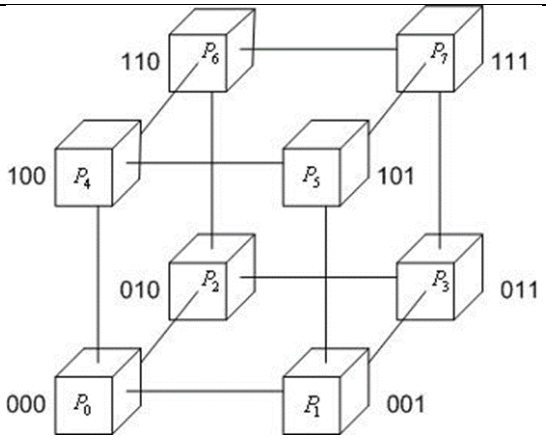
3) Расстояние между узлами коммуникационной сети – количество процессоров, которые образуют самый длинный путь между двумя узлами в графе коммуникационной сети. Диаметр коммуникационной сети – это минимальное расстояние между двумя произвольными ядрами этой сети.

11. Приведите примеры топологий коммуникационных сетей типа "шина", "линейка", "кольцо", "решетка", "клика", "бинарное дерево ", "звезда", "гиперкуб". Укажите диаметры *d* этих сетей.

(Найдите соответствие)

Топология типа «шина» (bus). *d=1*.	
Топология типа «линейка» (linear array). *d=N-1*	
Топология типа «кольцо» (ring). *d≈N/2*.	

Топология типа «двумерная решетка» (mesh). *d=sqrt(NM-1).*	
Топология типа «клика» (full connect). *d=1*.	
Топология типа «бинарное дерево» (tree). *d≈log*$_2$N.	
Топология типа «звезда» (star). *d=2.*	

Топология типа трехмерный «гиперкуб» (hypercube). Диаметр сети равен количеству измерений гиперкуба.	

12. Приведите классификацию сетевых коммутаторов.
(Найдите соответствие)

Простые сетевые коммутаторы с временным разделением	Коммутаторы представляют собой шины.
Простые сетевые коммутаторы с пространственным разделением	Коммутатор представляет собой совокупность мультиплексоров (МП), число которых равно количеству выходов коммутатора.
Составные сетевые коммутаторы	Коммутатор состоит из простых коммутаторов соединенных каналами типа «точка - точка».

13. Приведите структуру команды параллельной вычислительной машины потока данных и поясните смысл основных полей в этой структуре.

1) Структуру команды параллельной вычислительной машины потока данных имеет вид:

<Тэги><Адрес результата><Операнды><КОП>

Здесь Тэг – флаг готовности к обработке операнда, КОП – код операции. О готовности операции сигнализирует состояние «1» соответствующего кода. При установке всех кодов операции в состояние «1» команда переводится в состояние «готова к выполнению». На каждом такте машины потока данных готовые к исполнению операнды распределяются тэгами по процессорам.

+2) Структуру команды параллельной вычислительной машины потока данных имеет вид:

<КОП><Адрес результата><Тэги><Операнды>

Здесь КОП – код операции, каждому операнду соответствует флаг его готовности к обработке – тэг. О готовности операнда сигнализирует состояние «1» соответствующего тэга. При установке всех тэгов команды в состояние «1» команда переводится в состояние «готова к выполнению». На каждом такте машины потока данных готовые к исполнению команды распределяются коммутатором по процессорам.

3) Структуру команды параллельной вычислительной машины потока данных имеет вид:

<Операнды><Адрес результата><КОП><Тэги>

Здесь Операнд – код операции, каждой операции соответствует флаг его готовности к обработке – КОП. О готовности тэга сигнализирует состояние «1» соответствующего операнда. При установке всех операндов команды в состояние «1» команда переводится в состояние «простой». На каждом такте машины поток данных, готовых к исполнению команды распределяются коммуникатором по процессорам.

14. Назовите основные регистры ассоциативного запоминающего устройства и их функции.
(Найдите соответствие)

РгАП - регистр ассоциативных признаков	Из устройства управления передается бинарный вектор признаков искомой информации, имеющий от 1 до m разрядов.
РгМ - регистр маски	Если для выборки информации должен использоваться весь вектор признаков, то он без

	изменения поступает на схему сравнения; в противном случае - ненужные разряды этого вектора маскируются
РгИА - регистр индикаторов адреса	Если содержимое первого разряда ячейки с номером i не совпадает с содержимым первого разряда регистра ассоциативных признаков, то соответствующий этой ячейке разряд T_i устанавливается в состояние «0», в противном случае – «1».

15. Определите три основных класса операционных систем мультипроцессоров, их достоинства и недостатки.

(Найдите соответствие)

Каждый процессор имеет свою ОС.	Такая организация мультипроцессорных ОС использовалась на заре развития мультипроцессоров. В этом случае вся оперативная память статически разделяется на области по числу процессоров, каждому процессору выделяется своя область оперативной памяти и своя копия ОС, т.е. процессоры работают как независимые компьютеры.
ОС имеет только один процессор – "хозяин".	При таком способе организации операционной системы мультипроцессора используется процессором P_1 всего одна копия ОС. Процессоры P_2-P_N выполняют только пользовательские процессы. Процессор P_1, если у него есть свободное время, также может выполнять пользовательские процессы.
ОС может иметь любой процессор.	Здесь, в оперативной памяти находится всего одна копия операционной системы, но выполнять ее может любой из процессоров системы. При этом может быть обеспечена загрузка всех процессоров системы, возможно перераспределение памяти между процессорами и по сравнению с предыдущей организацией устраняется перегрузка процессора P_1.

16. Что такое расщепление операционной системы?

+1) Недостаток организации ОС в значительной мере преодолевается расщеплением операционной системы на такие части, как планирование вычислительных процессов, обращение к файловой системе, обработка страничных прерываний. В этом случае один процессор может выполнять ту часть операционной системы, которая осуществляет планирование процессов. В это же время другой процессор может выполнять файловую подсистему ОС и т.д. В этом случае, очевидно, также возможны взаимные блокировки, но степень распараллеливания работы ОС в общем случае выше.

2) Расщепление операционной системы – это процесс синхронизации параллельных процессов, когда определенное действие одного процесса должно быть выполнено после определенного действия другого процесса (например, один из процессов должен начаться после вычисления другим процессом значения некоторой переменной).

3) Единицы работы, между которыми операционная система МВС разделяет процессоры и другие ресурсы вычислительной системы, называется расщепление операционной системы. Любая работа вычислительной системы состоит в выполнении некоторой программы. Поэтому можно сказать, что расщепление операционной системы – это выполнение вычислительной системой некоторой системной или прикладной программы или их фрагмента.

17. Перечислите основные функции операционных систем многопроцессорных ВС.

(Внимание! Множественный выбор)

+1) функция синхронизации параллельных процессов;

2) функция проверки занятости процессами адресных каналов памяти;

+3) функция коммуникации параллельных процессов;

+4) функция управления распределенной памятью (в системах с распределенной памятью);

5) функция совместного использования процессорного пространства;

+6) функция планирования параллельных процессов.

18) Дайте определение вычислительного процесса.

+1) Единицы работы, между которыми операционная система МВС разделяет процессоры и другие ресурсы вычислительной системы, называется процессом. Любая работа вычислительной системы состоит в выполнении некоторой программы. Поэтому можно сказать, что процесс – это выполнение вычислительной системой некоторой системной или прикладной программы или их фрагмента.

2) Процесс в ОС в значительной мере представлен расщеплением операционной системы на такие части, как планирование вычислительных процессов, обращение к файловой системе, обработка страничных прерываний. В этом случае один процессор может выполнять ту часть операционной системы, которая осуществляет планирование процессов. В это же время другой процессор может выполнять файловую подсистему ОС и т.д. В этом случае, очевидно, также возможны взаимные блокировки, но степень распараллеливания работы ОС в общем случае выше.

3) Процесс – это синхронизация параллельных операндов, когда определенное действие одного операнда должно быть выполнено после определенного действия другого операнда (например, один из операндов должен начаться после вычисления другим операндом значения некоторой переменной).

19) Дайте определение контекста процесса.

(Внимание множественный выбор)

Каждому процессу в операционной системе соответствует контекст процесса, который включает в себя:

+1) пользовательский контекст (соответствующий программный код, данные, размер виртуальной памяти, дескрипторы открытых файлов и пр.).

2) сетевой контекст (сетевой протокол, сетевой интерфейс, межсетевой экран, протоколы взаимодействия сетей, сетевой идентификатор).

+3) аппаратный контекст (содержимое регистра счетчика команд, регистра состояния процессора, регистр указателя стека, а также содержимое регистров общего назначения).

+4) системный контекст (состояние процесса, идентификатор соответствующего пользователя, идентификатор процесса и пр.).

5) программный контекст (программный код, микропрограммная организация, размер виртуальных кодовых таблиц, микропрограммный код определения состояния процесса).

20. Назовите основные причины, по которым необходима синхронизация параллельных процессов.

1) Синхронизация параллельных процессов необходима в случае необходимости обеспечить заданную дисциплину доступа (последовательность доступа) к разделяемым ресурсам (например, к памяти, файлам и пр.)

2) Синхронизация параллельных процессов необходима в трех случаях:

а) когда определенное действие одного процесса должно быть выполнено после определенного действия другого процесса (например, один из процессов должен начаться после вычисления другим процессом значения некоторой переменной);

б) когда необходимо предотвратить проблемы, связанные с совместным использованием различными процессами разделяемых ресурсов, т.е. объявить запрет одновременного доступа к разделяемым ресурсам более чем одному процессу;

в) когда необходимо обеспечить заданную дисциплину доступа (последовательность доступа) к разделяемым ресурсам (например, к памяти, файлам и пр.).

+3) Синхронизация параллельных процессов необходима в двух случаях:

а) когда определенное действие одного процесса должно быть выполнено после определенного действия другого процесса (например, один из процессов должен начаться после вычисления другим процессом значения некоторой переменной);

б) когда необходимо обеспечить заданную дисциплину доступа (последовательность доступа) к разделяемым ресурсам (например, к памяти, файлам и пр.).

21. Дайте определение критической области (критической секции).

+1) Критической областью называется часть программы, в которой содержится обращение к разделяемому ресурсу. Это обращение является запретом одновременного доступа к разделяемым ресурсам более чем одному процессу, т.е. взаимное исключение.

2) Критическая область – это область взаимного исключения, где любые два процесса не могут одновременно находиться в одной области.

3) Критическая область – это область, в которой любое количество процессов может использовать попеременно данную секцию.

22. Определите понятия семафора.

+1) Семафор (*semaphore*) – это специальный тип переменных, которые могут принимать только неотрицательные значения, и над которыми определены только три следующие операции (*S* – переменная типа «семафор»):

а) *INIT*(*S*) – операция инициализации семафора;

б) операция *P*(*S*) – значение переменной *S* уменьшается на единицу, если это возможно; операция интерпретируется как операция опускания (закрытия) семафора;

в) операция *V*(*S*) – значение переменной *S* увеличивается на единицу; операция интерпретируется как операция поднятия (открытия) семафора.

2) Семафор (*semaphore*) – это процесс синхронизации параллельных операндов, когда определенное действие одного операнда должно быть выполнено после определенного действия другого операнда (например, один из операндов должен начаться после вычисления другим операндом значения некоторой переменной).

3) Семафор (*semaphore*) – это код операции, каждому операнду соответствует флаг его готовности к обработке. О готовности операнда сигнализирует состояние «1» соответствующего тэга. При установке всех семафоров команды в состояние «1» команда переводится в состояние «готова к выполнению». На каждом такте машины потока данных готовые к исполнению команды распределяются коммутатором по процессорам.

23. Определите понятия спин-блокировки и двоичного экспоненциального отката.

(Найдите соответствие)

Спин-блокировка	Состояние, в которое переходит процесс после достижения критической секции, занятой другим процессом.
Двоичный экспоненциальный откат	Между опросами вставляется цикл задержки. Вначале задержка равна времени выполнения одной команды. Если семафор занят, задержка удваивается, учетверяется и т.д. до некоторого максимального уровня. Этот уровень устанавливается, исходя из следующих соображений: низкий уровень обеспечивает высокую реактивность, но и высокую загрузку коммуникационной среды; высокий уровень, напротив, - низкую загрузку среды, но и низкую реактивность.

24. Назовите основные варианты реализации обмена данными между процессами посредством передачи сообщений.

(Внимание множественный выбор)

1) вызов **recive** возвращает управление вызывающему процессу до завершения получения сообщения.

+2) простое рандеву – операционная система мультикомпьютера предоставляет пользовательским программам средства отправки и получения сообщений в явном виде (в виде системных вызовов типа **send** и **receive**).

3) вызов **send** посылает сообщение указанному в вызове процессу и блокирует вызывающий процесс до завершения отправки сообщения.

+4) вызов удаленной процедуры – операционная система скрывает от пользователя передачу и прием сообщений под видом механизма вызова удаленной процедуры.

5) операция *P*(*S*) – значение переменной *S* уменьшается на единицу, если это возможно; операция интерпретируется как операция опускания (закрытия) семафора.

25. Какие функции по обмену данными между процессами происходят с помощью вызова удаленной процедуры (remote procedure call)?

(Внимание множественный выбор)

1) вызванная процедура процесса Q_I выполняется на процессоре P_j.

+2) процесс Q_I (процесс-клиент), выполняемый на процессоре P_I, вызывает процедуру (процесс-сервер Q_j), выполняемую на процессоре P_J и передает ей необходимые данные (параметры процедуры).

+3) процесс Q_I приостанавливается.

4) процесс Q_i посылает указанные результаты и приостанавливает выполнение.

+5) вызванная процедура выполняется на процессоре P_j.

+6) после завершения обработки полученных данных процесс-сервер Q_j передает процессу-клиенту Q_j результаты обработки.

7) при обработке полученных данных процессор P_j передает процессу-клиенту Q_j запрос на адресацию.

+8) процесс Q_i получает указанные результаты и продолжает выполняться.

26. Определите понятия простого рандеву с блокирующим вызовом.

(Внимание множественный выбор)

Обмен сообщениями между процессами по схеме простого рандеву с использование блокирующих вызовов происходит по следующей схеме:

+1) вызов **send** посылает сообщение указанному в вызове процессу и блокирует вызывающий процесс до завершения отправки сообщения.

2) вызов **recive** возвращает управление вызывающему процессу до завершения получения сообщения.

+3) вызов **recive** вызывает блокировку вызывающего процесса вплоть до завершения получения сообщения.

4) вызов **send** возвращает управление вызывающему процессу до завершения отправки сообщения.

27. Определите понятия простого рандеву с неблокирующим вызовом.

(Внимание множественный выбор)

Обмен сообщениями между процессами по схеме простого рандеву с использование неблокирующих вызовов происходит по следующей схеме:

+1) вызов **send** возвращает управление вызывающему процессу до завершения отправки сообщения.

2) вызов **send** посылает сообщение указанному в вызове процессу и блокирует вызывающий процесс до завершения отправки сообщения.

+3) вызов **recive** возвращает управление вызывающему процессу до завершения получения сообщения.

4) вызов **recive** вызывает блокировку вызывающего процесса вплоть до завершения получения сообщения.

28. Назовите преимущества модели обмена данными через общую память по сравнению с моделью обмена данными посредством передачи сообщений.

1) В ОС мультикомпьютеров используют алгоритм репликации, который расширяет миграционный алгоритм механизмом размножения страниц данных. Производительность при этом повышается за счет возможности одновременного доступа к данным по чтению. Недостатком алгоритма является необходимость отслеживания расположение всех страниц данных и их копий, а также значительные затраты на уничтожение всех устаревших копий страниц или их коррекции.

+2) В ОС мультикомпьютеров обычно используют модель обмена данными между процессами посредством передачи сообщений. Альтернативой является модель обмена данными через общую память – DSM-модель обмена данными (Distributed Shared Memory). В программе, использующей для обмена данными модель передачи сообщений, программист обеспечивает доступ к данным посредством явных операций посылки и приема сообщений. Это обстоятельство сильно усложняет программирование и отладку программы. DSM-модель обмена данными скрывает от программиста пересылку данных и обеспечивает ему абстракцию разделяемой памяти (привычной для него по опыту работы на традиционных ЭВМ и мультипроцессорах). В результате программирование и отладка с использованием DSM-модели обмена данными становится гораздо проще.

3) В ОС мультикомпьютеров применяется обмен информацией о своей текущей загрузке на каждой итерации решения задачи, периодически, когда загрузка процессора станет меньше некоторой допустимой, по появлении свободного процессора, по отторжении процесса процессором вследствие перегрузки и т.д. Решение о перераспределении загрузки может приниматься централизованно (на основе глобальной информации о загрузке) и децентрализовано (на основе только локальной информации о загрузке).

29. Назовите основные функции подсистемы управления распределенной памятью в ОС мультикомпьютеров.

+1) Основной целью управления распределенной оперативной памятью мультикомпьютера является создание виртуального адресного пространства, разделяемое всеми его процессорами. Процессы, выполняемые на разных процессорах, должны получать доступ к данным в распределенной памяти системы примерно так же, как это происходит с данными в виртуальной памяти традиционных ЭВМ.

2) Процесс управления распределенной памятью в ОС, который достиг критической секции, занятой другим процессом, переходит в состояние спин-блокировки. В это время он непрерывно и с максимальной скоростью опрашивает состояние семафора, что приводит, во-первых, к напрасной трате времени соответствующего процессора, а, во-вторых, накладывает значительную нагрузку на коммуникационную сеть и память, снижая тем самым скорость работы остальных процессоров

3) Процессоры могут обмениваться информацией в ОС о своей текущей загрузке на каждой итерации решения задачи, периодически, когда загрузка процессора станет меньше некоторой допустимой, по появлении свободного процессора, по отторжении процесса процессором вследствие перегрузки и т.д.

30. Какие элементы входят в схему алгоритма управления распределенной памятью?

(Внимание множественный выбор)

+1) если процесс Q_i запрашивает страницу памяти, которая отсутствует в оперативной памяти данного процессора P_i, то происходит страничное прерывание с передачей управления операционной системе;

2) после завершения обработки полученных данных процесс-сервер Q_j передает процессу-клиенту Q_j результаты обработки;

+3) операционная система находит необходимую страницу в оперативной памяти процессора P_i и требует от него переслать эту страницу процессору P_i;

+4) процессор P_j передает по коммуникационной сети требуемую страницу в оперативную память процессора P_i;

5) вызов удаленной процедуры – операционная система скрывает от пользователя передачу и прием сообщений под видом механизма вызова удаленной процедуры;

+6) команда, вызвавшая страничное прерывание, перезапускается (на процессоре P_i).

31. Поясните суть, а также назовите основные достоинства и недостатки алгоритма репликации для управления распределенной памятью.

+1) Достоинством алгоритма репликации является повышение производительности доступа к распределенной памяти за счет возможности одновременного доступа к данным по чтению, а также расширение миграционного алгоритма механизмом размножения страниц данных.

Недостатком алгоритма репликации является необходимость отслеживания расположение всех страниц данных и их копий, а также значительные затраты на уничтожение всех устаревших копий страниц или их коррекции.

2) Достоинством алгоритма репликации является то, что любые два процесса не могут одновременно находиться в одной критической области.

Недостатком алгоритма репликации является процесс, находящийся вне данной критической области, не может блокировать доступ других процессов к этой области.

3) Достоинством алгоритма репликации является то, что ни одни из процессов не может занимать критическую область бесконечно долго.

Недостатком алгоритма репликации является попеременное использование критической секции, которая может использовать любое количество процессов.

32. Поясните суть алгоритма полного размножения для управления распределенной памятью.

1) Алгоритм полного размножения выполняет балансировку загрузки, при которой в процессе решения задачи производится перераспределение процессов по процессорам. Алгоритм полного размножения приходится использовать в том случае, когда в процессе вычислений процессы $Q_1, Q_2, \ldots, Q_n$ выполняются многократно и вычислительная сложность их меняется от итерации к итерации. Статическая балансировка загрузки при этом может быть неэффективной.

+2) Алгоритм полного размножения для управления распределенной памятью является расширением алгоритма репликации и заключается в репликации всех страниц. Алгоритм позволяет многим процессорам иметь одновременный доступ к разделяемым данным на чтение и запись. Поскольку процессоры могут записывать данные параллельно, возникает проблема обеспечения консистентности данных.

3) Алгоритм полного размножения осуществляется посредством планировщика операционной системы мультипроцессора, который призван решать следующие основные задачи планирования процессов: какой из готовых к выполнению процессов запустить в данный момент времени; на каком процессоре запустить этот процесс.

33. Определите понятия статической балансировки загрузки и динамической балансировки загрузки МВС. (Найдите соответствие)

Статическая балансировка загрузки	Балансировка загрузки, при которой распределение процессов по процессорам не меняется во время решения задачи.
Динамическая балансировка загрузки	Балансировка загрузки, при которой в процессе решения задачи производится перераспределение процессов по процессорам.

34. Назовите основные задачи, которые призван решать планировщик процессов операционной системы мультипроцессора.

+1) Планировщик операционной системы мультипроцессора призван решать две следующие основные задачи планирования процессов:

а) какой из готовых к выполнению процессов запустить в данный момент времени;

б) на каком процессоре запустить этот процесс.

2) Планировщик операционной системы мультипроцессора призван решать три следующие основные задачи планирования процессов:

а) создание виртуального адресного пространства, разделяемое всеми его процессорами;

б) вызов удаленной процедуры – операционная система скрывает от пользователя передачу и прием сообщений под видом механизма вызова удаленной процедуры;

в) перезапуск команды, вызвавшей страничное прерывание.

3) Планировщик операционной системы мультипроцессора призван решать две следующие основные задачи планирования процессов

а) вызов процедуры **send**, которая возвращает управление вызывающему процессу до завершения отправки сообщения;

б) вызов процедуры **recive**, которая вызывает блокировку вызывающего процесса вплоть до завершения получения сообщения.

.

5.4 Вопросы к экзамену или зачету

1. Классификация вычислительных систем Флинна. Основные классификационные признаки.
2. Типы вычислительных систем по классификации Флинна.
3. Основные классы параллельных вычислительных систем в зависимости от типа строения их оперативной памяти.
4. SIMD-вычислительные системы. Основные классы.
5. MIMD-вычислительные системы. Основные классы.
6. Вычислительные кластеры. Основные классификационные признаки.
7. Гипотеза Минского. Основные причины невыполнения ее в современных параллельных вычислительных системах.
8. Топологии коммуникационных сетей.
9. Классификация сетевых коммутаторов.
10. Структура команды параллельной вычислительной машины потока данных.

11. Структура ассоциативного запоминающего устройства.
12. Основные классы операционных систем мультипроцессоров.
13. Что такое расщепление операционной системы?
14. Синхронизация параллельных процессов. Основные причины.
15. Определите понятия семафора и двоичного семафора.
16. Схема обмена данными между процессами с помощью вызова удаленной процедуры.
17. Понятие простого рандеву с блокирующим вызовом.
18. Понятие простого рандеву с неблокирующим вызовом.
19. Функции подсистемы управления распределенной памятью в ОС мультикомпьютеров.
20. Миграционный алгоритм управления распределенной памятью. Схема. Основные достоинства и недостатки.
21. Статическая и динамическая балансировки загрузки МВС.
22. Планировщик процессов операционной системы мультипроцессора. Основные задачи.
23. Методы динамической балансировки загрузки процессора.
24. Алгоритм планирования независимых процессов с использованием очередей заданий.
25. Простейший алгоритм планирования зависимых процессов.
26. Отображение процессов с регулярной структурой на типовые архитектуры мультикомпьютеров.
27. Выборка элементов массива.
28. Функции обработки массивов для получения соответствия.
29. Параллельные операторы.
30. Языки высокого уровня для программирования MIMD-вычислительных систем.
31. Автоматическое распараллеливание последовательных программ.
32. Коммуникационные библиотеки для организации параллельного выполнения программ.
33. Средства отладки и профилирования параллельных программ.
34. Типы параллелизма и методы синтеза параллельных алгоритмов.
35. Оценка эффективности параллельных алгоритмов.
36. Параллельные алгоритмы вычисления рекурсий.
37. Базовые алгоритмы линейной алгебры.
38. Прямые методы решения систем линейных алгебраических уравнений. Метод исключения Гаусса
39. Итерационные методы решения систем линейных алгебраических уравнений. Метод Якоби
40. Методы минимизации для решения систем линейных алгебраических уравнений. Метод сопряженных направлений
41. Параллельные алгоритмы интегрирования обыкновенных дифференциальных уравнений.
42. Двухточечная краевая задача для уравнений второго порядка.
43. Параллельные алгоритмы решения краевых задач для дифференциальных уравнений в частных производных.
44. Нестационарная задача для дифференциальных уравнений в частных производных.
45. Стационарная краевая задача для дифференциальных уравнений в частных производных.