

**ФЕДЕРАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЦЕНТР
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ ИМ. М.В. КЕЛДЫША
РОССИЙСКОЙ АКАДЕМИИ НАУК**

Утверждена

Ученым советом ФИЦ ИПМ

им. М.В. Келдыша РАН,

протокол № __ от «__» _____ 2016_

г.

Заместитель директора

_____ А.Л. Афендииков

(подпись, расшифровка подписи)

«__» _____ 2016 г.

РАБОЧАЯ ПРОГРАММА

УЧЕБНОЙ ДИСЦИПЛИНЫ

Операционные системы

Направление подготовки

09.06.01 – «Информатика и вычислительная техника»

Профили (направленности программы)

05.13.11– «Математическое обеспечение вычислительных машин, комплексов и компьютерных сетей»

Квалификация выпускника

Исследователь. Преподаватель-исследователь

Форма обучения

очная

Москва, 2016

Направление подготовки: 09.06.01 – «Информатика и вычислительная техника»

Профиль (направленность программы): 05.13.11 – «Математическое обеспечение вычислительных машин, комплексов и компьютерных сетей»

Дисциплина: Операционные системы

Форма обучения: очная

Рабочая программа составлена с учетом ФГОС ВО по направлению подготовки 09.06.01 – «Информатика и вычислительная техника», утвержденного приказом Министерства образования и науки Российской Федерации от 30 июля 2014 г. N 875, зарегистрировано в Минюсте Российской Федерации 20 августа 2014 г. N 33685, и Программы-минимум кандидатского экзамена по специальности, утвержденной приказом Министерства образования и науки Российской Федерации от 8 октября 2007 года № 274 (зарегистрировано Минюстом Российской Федерации 19 октября 2007 года № 10363).

РЕЦЕНЗЕНТ: Крюков Виктор Алексеевич, ФИЦ ИПМ им. М.В. Келдыша РАН, заведующий отделом, доктор физико-математических наук, профессор

РАБОЧАЯ ПРОГРАММА РЕКОМЕНДОВАНА

Ученым советом ФИЦ ИПМ им. М.В. Келдыша РАН, протокол № ___ от «___» _____ 2016 г.
Заместитель директора А.Л. Афендииков.

ИСПОЛНИТЕЛЬ (разработчик программ):

Бахтин В.А., ФИЦ ИПМ им. М.В. Келдыша РАН, заведующий сектором, к.ф.-м.н.

Рабочая программа зарегистрирована в аспирантуре под учетным номером _____ на правах учебно-методического издания.

Заведующий аспирантурой _____ / Меньшов И.С. /

Оглавление

АННОТАЦИЯ	4
1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ	4
2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ	4
3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ.....	5
3.1. Структура дисциплины	5
3.2. Содержание разделов дисциплины	6
3.3. Лекционные занятия	8
3.4. Семинарские занятия	8
4. ТЕКУЩАЯ, ПРОМЕЖУТОЧНАЯ И ИТОГОВАЯ АТТЕСТАЦИЯ. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ.....	9
5. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	16
6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ.....	17

АННОТАЦИЯ

Дисциплина «Операционные системы» реализуется в рамках Блока 1 Основной профессиональной образовательной программы высшего образования – программы подготовки научно-педагогических кадров в аспирантуре Федерального исследовательского центра Института прикладной математики им. М.В. Келдыша РАН (ИПМ им. М.В. Келдыша РАН) по направлению подготовки 09.06.01 – «Информатика и вычислительная техника».

Рабочая программа разработана с учетом требований ФГОС ВО по направлению подготовки 09.06.01 – «Информатика и вычислительная техника», утвержденного приказом Министерства образования и науки Российской Федерации от 30 июля 2014 г. N 875, зарегистрировано в Минюсте Российской Федерации 20 августа 2014 г. N 33685, и Программы-минимум кандидатского экзамена по специальности, утвержденной приказом Министерства образования и науки Российской Федерации от 8 октября 2007 года № 274 (зарегистрировано Минюстом Российской Федерации 19 октября 2007 года № 10363).

Основным источником материалов для формирования содержания программы являются: материалы конференций, симпозиумов, семинаров, Интернет-ресурсы, научные издания и монографические исследования и публикации.

Общая трудоемкость дисциплины по учебному плану составляет 2 зач.ед. (72 часа), из них лекций – 4 часа, семинарских занятий – 10 часов, практических занятий – 0 часов, самостоятельной работы – 22 часа, подготовка к экзаменам – 36 часов. Дисциплина реализуется на 1-м курсе, во 2-м семестре, продолжительность обучения – 1 семестр.

Текущая аттестация проводится не менее 2 раз в соответствии с заданиями и формами контроля, предусмотренными настоящей программой.

Итоговая оценка знания осуществляется в период зачетно-экзаменационной сессии в форме экзамена.

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цели и задачи дисциплины «Операционные системы»

Цель:

- получение фундаментальных знаний о направлениях развития, принципах построения и функционирования современных операционных систем.

Задачи:

- владение математическим аппаратом, позволяющим выбрать наиболее подходящий механизм или эффективный алгоритм, согласно критериям проблемной области;
- практическое освоение накопленных по дисциплине знаний при решении профессиональных проблем в реальных (смоделированных) условиях;
- стимулирование к самостоятельной деятельности по освоению дисциплины и формированию необходимых компетенций.

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины «Операционные системы» направлен на формирование компетенций или отдельных их элементов в соответствии с ФГОС ВО по направлению подготовки 09.06.01 – «Информатика и вычислительная техника», утвержденного приказом Министерства образования и науки Российской Федерации от 30 июля 2014 г. N 875, зарегистрировано в Минюсте Российской Федерации 20 августа 2014 г. N 33685

а) универсальные (УК): не предусмотрено;

б) общепрофессиональные (ОПК): не предусмотрено;

в) профессиональные (ПК): Способность использовать языки программирования и системы программирования (ПК-1), Способность создавать модели и алгоритмы проектирования программных систем (ПК-2).

В результате освоения дисциплины обучающийся должен:

Знать:

- классификацию ОС;
- назначение и функции ОС;
- логику функционирования ОС;
- характеристики современных ОС;
- принципы работы основных подсистем ОС;
- основные механизмы управления ресурсами вычислительной системы.

Уметь:

- использовать различные механизмы и алгоритмы для синхронизации процессов/нитей при доступе к разделяемой памяти, к общей файловой системе;
- реализовать коллективные MPI-операции рассылки/сбора данных при помощи сообщений точка-точка.

Владеть:

- основными средствами для межпроцессного взаимодействия;
- навыками оценки времени работы различных алгоритмов (выбора координатора, прохождения критических секций, надежных и неделимых широкоэвентельных рассылок сообщений, фиксации консистентного и строго консистентного множества контрольных точек), используемых в распределенных операционных системах, при заданных параметрах - количестве ЭВМ и характеристиках сети (времени старта и времени передачи байта);
- навыками оценки времени выполнения операций модификации/чтения переменных из распределенной общей памяти (DSM) для заданной модели консистентности памяти и при заданных параметрах сети (времени старта и времени передачи байта).

Приобрести опыт:

- работы в различных операционных системах;
- использования технологий OpenMP и MPI.

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

3.1. Структура дисциплины

Распределение трудоемкости дисциплины по видам учебных работ

Вид учебной работы	Трудоемкость	
	общая	
	зач.ед.	час.
ОБЩАЯ ТРУДОЕМКОСТЬ по Учебному плану	2	72
Лекции (Л)		4

Практические занятия (ПЗ)	-	-
Семинары (С)		10
Самоподготовка (проработка и повторение лекционного материала и материала учебников и учебных пособий, подготовка к семинарским и практическим занятиям) и самостоятельное изучение тем дисциплины		58
Вид контроля: экзамен		

3.2. Содержание разделов дисциплины

Общее содержание дисциплины

№ раздела	Наименование раздела	Содержание раздела	Форма текущей аттестации
1.	Введение в операционные системы	История операционных систем (ОС). Типы ОС: пакетные ОС, системы разделения времени, системы реального времени. Виды ОС: ОС мультимикропроцессорных ЭВМ, сетевые ОС, распределенные ОС. Обзор аппаратного обеспечения компьютера: многоядерные и многопоточные процессоры, память, диски, устройства ввода/вывода, шины. Основные функции ОС. Структура ОС: монолитные системы, многоуровневые системы, микроядра, клиент-серверная модель, виртуальные машины.	О, ДЗ
2.	Процессы и нити	Управление процессами. Определение процесса. Жизненный цикл, состояния процесса. Основные типы процессов: полновесные процессы, легковесные процессы (нити). Контекст процесса. Планирование распределения времени ЦП между процессами - основные подходы: вытесняющие и невытесняющие стратегии; алгоритмы, основанные на квантовании; алгоритмы, использующие приоритет: планирование по наивысшему приоритету, понятие относительного и абсолютного приоритета, алгоритмы, использующие изменяющийся приоритет. Очереди с обратной связью. Смешанные алгоритмы планирования.	О, ДЗ
3.	Взаимодействие процессов	Взаимодействие параллельных процессов и их синхронизация. Разделяемые ресурсы и синхронизация доступа к ним. Взаимное исключение. Тупики. Механизмы обеспечения взаимного исключения и координации: семафоры Дейкстры, мьютексы, механизм событий, обмен сообщениями. Алгоритм Деккера. Алгоритм Петерсона. Классические	

		задачи синхронизации процессов: «обедающие философы», «читатели и писатели», «спящий парикмахер». Синхронизация в распределенных системах: синхронизация времени, выбор координатора. Типовые средства межпроцессного взаимодействия (базовые средства взаимодействия процессов ОС Unix, IPC ОС Unix, сокеты). Технологии MPI и OpenMP.	
4.	Управление оперативной памятью	Базовые концепции и стратегии управления оперативной памятью. Виртуальная оперативная память. Организация управления памятью при одиночном непрерывном распределении; распределении разделами: распределении перемещаемыми разделами; страничном распределении (таблица страниц. TLB, иерархическая организация таблицы страниц, хэширование таблицы страниц, инвертированные таблицы страниц, алгоритмы замещения страниц); сегментном распределении; сегментно-страничном распределении. Кэширование данных при управлении оперативной памятью. Управление оперативной памятью в распределенных вычислительных системах, распределенная общая память (DSM), алгоритмы реализации, модели консистентности.	О, ДЗ
5.	Файловые системы	Файлы, структурная организация файлов. Атрибуты файлов. Типовые программные интерфейсы работы с файлами. Структура файловой системы, подходы в практической реализации. Модели реализации файлов. Понятие индексного узла (дескриптора). Модели реализации каталогов. Квотирование пространства файловой системы. Надежность файловой системы, целостность файловой системы. Семантика одновременного доступа к одному файлу нескольких процессов. Организация файловой системы ОС Unix. Распределенные и параллельные файловые системы. Алгоритмы обеспечения консистентности кэшей в распределенных файловых системах. Способы организации размножения файлов и коррекции копий. Файловые системы NFS, HDFS, GFS, Lustre.	О, ДЗ
6.	Управление внешними	Устройства ввода-вывода. Контроллеры устройств. Программное управление	О, ДЗ

	устройствами. Ввод и вывод информации	внешними устройствами. Драйверы физических и логических устройств. Обработка прерываний. Жесткие диски: форматирование, алгоритмы планирования перемещения блока головок, обработка ошибок, RAID-системы. Таймер: устройство, программное обеспечение. Пользовательский интерфейс: клавиатура, мышь, монитор.	
7.	Обеспечение надежности	Отказы в распределенных системах. Сообщения-сироты и эффект домино. Потеря сообщений. Проблема бесконечного восстановления. Консистентное множество контрольных точек. Алгоритмы фиксации контрольных точек и восстановления после отказа. Протоколы голосования. Алгоритм надежных и неделимых широковещательных рассылок сообщений. Отказоустойчивость.	О, ДЗ

Примечание: О – опрос, Д – дискуссия (диспут, круглый стол, мозговой штурм, ролевая игра), ДЗ – домашнее задание (эссе и пр.). Формы контроля не являются жесткими и могут быть заменены преподавателем на другую форму контроля в зависимости от контингента обучающихся. Кроме того, на занятиях семинарских может проводиться работа с нормативными документами, изданиями средств информации и прочее, что также оценивается преподавателем.

3.3. Лекционные занятия

№ занятия	№ раздела	Краткое содержание темы занятия	Кол-во часов
1.	3	Взаимодействие параллельных процессов и их синхронизация. Разделяемые ресурсы и синхронизация доступа к ним. Взаимное исключение. Тупики. Механизмы обеспечения взаимного исключения и координации: семафоры Дейкстры, мьютексы, механизм событий, обмен сообщениями. Способы реализации взаимного исключения в однопроцессорных ЭВМ, многопроцессорных ЭВМ с общей памятью, в распределенных вычислительных системах. Классические задачи синхронизации процессов: «обедающие философы», «читатели и писатели», «спящий парикмахер».	2
2.	5	Файлы, структурная организация файлов. Атрибуты файлов. Типовые программные интерфейсы работы с файлами. Распределенные и параллельные файловые системы. Алгоритмы обеспечения консистентности кэшей в распределенных файловых системах. Примеры файловых систем: NFS, HDFS.	2
ВСЕГО			4

3.4. Семинарские занятия

№ занятия	№ раздела	Краткое содержание темы занятия	Кол-во часов
1.	3	Задачи по теме взаимодействие процессов на	2

		многопроцессорных ЭВМ с общей памятью.	
2.	3	Задачи по теме взаимодействие процессов в распределенных вычислительных системах.	2
3.	3	Задачи по темам: технологии MPI и OpenMP.	2
4.	4	Задачи по темам: распределенная общая память (DSM), алгоритмы реализации, модели консистентности.	2
5.	7	Задачи по теме обеспечение надежности.	2
ВСЕГО			10

4. ТЕКУЩАЯ, ПРОМЕЖУТОЧНАЯ И ИТОГОВАЯ АТТЕСТАЦИЯ. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Текущая и промежуточная аттестация аспирантов. Текущая и промежуточная аттестация аспирантов проводится в соответствии с локальным актом ФИЦ ИПМ им. М.В. Келдыша РАН - Положением о текущей, промежуточной и итоговой аттестации аспирантов ФИЦ ИПМ им. М.В. Келдыша РАН по программам высшего образования – программам подготовки научно-педагогических кадров в аспирантуре и является обязательной.

Текущая аттестация по дисциплине проводится в форме опроса, а также оценки вопроса-ответа в рамках участия обучающихся в дискуссиях и различных контрольных мероприятиях, осуществляемых преподавателем, ведущим дисциплину. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины см. ниже.

Объектами оценивания выступают:

- учебная дисциплина – активность на занятиях, своевременность выполнения различных видов заданий, посещаемость занятий;
- степень усвоения теоретических знаний и уровень овладения практическими умениями и навыками по всем видам учебной работы, проводимых в рамках семинаров, практических занятий и самостоятельной работы.

Оценивание обучающегося на занятиях осуществляется с использованием нормативных оценок по 4-х бальной системе (5-отлично, 4-хорошо, 3-удовлетворительно, 2-не удовлетворительно).

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Форма контроля знаний	Вид аттестации	Примечание
Проверочные работы в течение всего курса	Текущая	Ниже приведен перечень рекомендуемых контрольных вопросов
Проверочные работы в течение всего курса	Промежуточная	Ниже приведен перечень рекомендуемых задач и контрольных вопросов

Примерный перечень рекомендуемых контрольных вопросов для оценки текущего уровня успеваемости аспиранта:

1. В чем заключаются основные функции операционной системы?
2. В чем разница между системами с разделением времени и многозадачными системами?
3. В чем разница между режимом ядра и пользовательским режимом?
4. В современных операционных системах адресное пространство процесса отделено от физической памяти машины. Назовите преимущества такой реализации.

5. Какие аппаратные механизмы необходимы для организации мультипрограммного режима? Как обеспечить мультипрограммный режим без этих механизмов? Как обеспечить, если отсутствует только один из этих механизмов?
6. В чем отличие между легковесными и полновесными процессами?
7. Перечислите основные алгоритмы для планирования распределения времени ЦП между процессами.
8. Какие механизмы для синхронизации процессов/нитей при доступе к разделяемой памяти существуют?
9. Что такое файловый сервер и файловый сервис?
10. Что такое дескриптор файла?
11. Какие существуют подходы к именованию файлов?
12. Существуют файловые серверы с состоянием и без состояния. Перечислите их достоинства и недостатки.
13. Какие алгоритмы используются в распределенных файловых системах для обеспечения консистентности кэшей?
14. Какие алгоритмы применяются в распределенных файловых системах для организации размножения файлов и коррекции копий?
15. Что такое когерентность и консистентность памяти?
16. Какие модели консистентности памяти существуют?
17. Какие модели консистентности памяти удовлетворяют алгоритму Деккера (алгоритм без каких-либо изменений будет работать правильно), а какие нет? Объясните ответ.
18. Какие модели консистентности памяти удовлетворяют алгоритму Петерсона (алгоритм без каких-либо изменений будет работать правильно), а какие нет? Объясните ответ.
19. Какие режимы передачи сообщений точка-точка в MPI существуют?
20. Какие средства для синхронизации нитей в OpenMP существуют?

Примерный перечень рекомендуемых контрольных задач для оценки промежуточного уровня успеваемости аспиранта:

1. Имеется механизм двоичных семафоров. Опираясь на него, реализуйте P-операцию и V-операцию для общего (считающего) семафора. Активное ожидание освобождения семафора не допускается.
2. Имеется команда TSL и команда объявления прерывания указанному процессору. Опираясь на них, реализуйте на мультипроцессоре P-операцию и V-операцию для двоичного семафора. Активное ожидание освобождения семафора не допускается.
3. Имеется механизм двоичных семафоров. Опираясь на него, реализуйте операторы POST(имя переменной-события) и WAIT(имя переменной-события). Активное ожидание события не допускается. Оцените, во сколько раз нижеприведенный алгоритм метода последовательной верхней релаксации можно выполнить быстрее, чем последовательный, если число процессоров мультипроцессора = N, время выполнения одного оператора присваивания ($A[i][j]=\dots$) равно 1, временами выполнения остальных операторов можно пренебречь.

```
float A[ L1 ][ L2 ];
```

```
/* массив двоичных семафоров с нулевым начальным значением */
```

```
semaphore s[ L1 ][ L2 ];
```

```
for ( j = 0; j < L2; j++ ) { post( s[ 0 ][ j ] ) }
```

```
parfor ( i = 1; i < L1-1; i++ )
```

```
  for ( j = 1; j < L2-1; j++ )
```

```
    { wait( s[ i-1 ][ j ] );
```

$$A[i][j] = (A[i-1][j] + A[i+1][j] + A[i][j-1] + A[i][j+1]) / 4;$$

post(s[i][j]);

}

4. Имеется механизм двоичных семафоров. Опираясь на него, реализуйте задачу читателей и писателей (алгоритмы предоставления прав доступа процессам-читателям и процессам-писателям): Процесс-писатель должен получать исключительный (монопольный) доступ к базе данных (других писателей или каких-либо читателей быть не должно). Произвольное число процессов-читателей может работать одновременно, но любой читатель может получить доступ только при отсутствии работающих писателей. Запросы на доступ должны удовлетворяться "справедливо" - в порядке их поступления (можно исходить из "справедливости" удовлетворения запросов на двоичные семафоры).
5. В транспьютерной матрице размером 4×4 , в каждом узле которой находится один процесс, необходимо выполнить операцию передачи сообщения длиной N байт всем процессам от одного (MPI_BCAST) - процесса с координатами $(0,0)$. Сколько времени потребуется для этого, если все процессы выдали ее одновременно. Время старта равно 100 , время передачи байта равно 1 ($T_s=100, T_b=1$). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
6. В транспьютерной матрице размером 4×4 , в каждом узле которой находится один процесс, необходимо выполнить операцию сбора данных (длиной один байт) от всех процессов для одного (MPI_GATHER) - процесса с координатами $(0,0)$. Сколько времени потребуется для этого, если все процессы выдали ее одновременно. Время старта равно 100 , время передачи байта равно 1 ($T_s=100, T_b=1$). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
7. В транспьютерной матрице размером 4×4 , в каждом узле которой находится один процесс, необходимо выполнить операцию рассылки данных (длиной один байт) всем процессам от одного (MPI_SCATTER) - процесса с координатами $(0,0)$. Сколько времени потребуется для этого, если все процессы выдали ее одновременно. Время старта равно 100 , время передачи байта равно 1 ($T_s=100, T_b=1$). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
8. В транспьютерной матрице размером 4×4 , в каждом узле которой находится один процесс, необходимо выполнить операцию нахождения максимума среди 16 чисел (каждый процесс имеет свое число). Сколько времени потребуется для получения всеми максимального числа, если все процессы выдали эту операцию редукации одновременно. А сколько времени потребуется для нахождения максимума среди 64 чисел в матрице 8×8 ? Время старта равно 100 , время передачи байта равно 1 ($T_s=100, T_b=1$). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
9. В транспьютерной матрице размером 4×4 , в каждом узле которой находится один процесс, необходимо переслать очень длинное сообщение (длиной L байт) из узла с координатами $(0,0)$ в узел с координатами $(3,3)$. Сколько времени потребуется для этого, если передача сообщений точка-точка выполняется в буферизируемом режиме MPI? А сколько времени потребуется при использовании синхронного режима и режима готовности? Время старта равно 100 , время передачи байта равно 1 ($T_s=100, T_b=1$). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
10. В транспьютерной матрице размером 4×4 , в каждом узле которой находится один процесс, необходимо переслать сообщение длиной L байт из узла с координатами $(0,0)$ в узел с координатами $(3,3)$. Сколько времени потребуется для этого при использовании а) неблокирующих и б) блокирующих операций MPI? Время

- старта равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
11. Все 16 процессов, находящихся на разных ЭВМ сети с шинной организацией (без аппаратных возможностей широковещания), одновременно выдали запрос на вход в критическую секцию. Сколько времени потребуется для прохождения всеми критических секций, если используется древовидный маркерный алгоритм (маркером владеет нулевой процесс). Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
 12. Все 16 процессов, находящихся на разных ЭВМ сети с шинной организацией (без аппаратных возможностей широковещания), одновременно выдали запрос на вход в критическую секцию. Сколько времени потребуется для прохождения всеми критических секций, если используется децентрализованный алгоритм с временными метками. Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
 13. Все 16 процессов, находящихся на разных ЭВМ сети с шинной организацией (без аппаратных возможностей широковещания), одновременно выдали запрос на вход в критическую секцию. Сколько времени потребуется для прохождения всеми критических секций, если используется широковещательный маркерный алгоритм (маркером владеет нулевой процесс). Время старта равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
 14. 15 процессов, находящихся в узлах транспьютерной матрицы размером 4×4 , одновременно выдали запрос на вход в критическую секцию. Сколько времени потребуется для прохождения всеми критических секций, если используется централизованный алгоритм (координатор расположен в узле 0,0)? Время старта равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
 15. Сколько времени потребует выбор координатора среди 16 процессов, находящихся на разных ЭВМ сети с шинной организацией (без аппаратных возможностей широковещания), если используется алгоритм "задиры"? "Задира" расположен в узле с координатами (0,0) и имеет уникальный номер 0. Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
 16. Сколько времени потребует выбор координатора среди 16 процессов, находящихся в узлах транспьютерной матрицы размером 4×4 , если используется круговой алгоритм? Время старта равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Процессорные операции, включая чтение из памяти и запись в память считаются бесконечно быстрыми.
 17. Последовательная консистентность памяти и алгоритм ее реализации в DSM с полным размножением. Сколько времени потребует модификация 10 различных

переменных 10-ю процессами (каждый процесс модифицирует одну переменную), находящимися на разных ЭВМ сети с шинной организацией (без аппаратных возможностей широко вещания) и одновременно выдавшими запрос на модификацию. Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.

18. Причинная консистентность памяти и алгоритм ее реализации в DSM с полным размножением при условии, что никаких сведений от компилятора о причинной зависимости операций записи не имеется. Сколько времени потребует модификация 10 различных переменных, если все 10 процессов (каждый процесс модифицирует одну переменную), находящихся на разных ЭВМ сети с шинной организацией (без аппаратных возможностей широко вещания), одновременно выдали запрос на модификацию своей переменной. Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
19. Процессорная консистентность памяти и алгоритм ее реализации в DSM с полным размножением. Сколько времени потребует модификация 10 различных переменных, если все 10 процессов (каждый процесс модифицирует одну переменную), находящихся на разных ЭВМ сети с шинной организацией (без аппаратных возможностей широко вещания), одновременно выдали запрос на модификацию своей переменной. Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
20. PRAM консистентность памяти и алгоритм ее реализации в DSM с полным размножением. Сколько времени потребует 3-кратная модификация 10 различных переменных, если все 10 процессов (каждый процесс 3 раза модифицирует одну переменную), находящихся на разных ЭВМ сети с шинной организацией (без аппаратных возможностей широко вещания), одновременно выдали запрос на модификацию. Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
21. Слабая консистентность памяти и алгоритм ее реализации в DSM с полным размножением. Сколько времени потребует модификация одним процессом 10 обычных переменных, а затем 3-х различных синхронизационных переменных, если DSM реализована на 10 ЭВМ сети с шинной организацией (с аппаратными возможностями широко вещания). Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.

22. Консистентность памяти по выходу и алгоритм ее реализации в DSM с полным размножением. Сколько времени потребует трехкратное выполнение каждым процессом критической секции, в которой модифицируются 10 переменных, если DSM реализована на 10 ЭВМ сети с шинной организацией (с аппаратными возможностями широковещания). Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
23. Консистентность памяти по входу и алгоритм ее реализации в DSM с полным размножением. Сколько времени потребует трехкратное выполнение критической секции и модификация в ней 10 переменных каждым процессом, если DSM реализована на 10 ЭВМ сети с шинной организацией (с аппаратными возможностями широковещания). Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память считаются бесконечно быстрыми.
24. Алгоритм надежных и неделимых широковещательных рассылок сообщений. Дайте оценку времени выполнения одной операции рассылки для сети из 10 ЭВМ с шинной организацией (без аппаратных возможностей широковещания), если отправитель сломался после посылки 5-го сообщения. Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
25. Протоколы голосования. Дайте оценку времени выполнения одним процессом 2-х операций записи и 10 операций чтения N байтов информации с файлом, расположенным (размноженным) на остальных 10 ЭВМ сети с шинной организацией (без аппаратных возможностей широковещания). Определите оптимальные значения кворума чтения и кворума записи для $N=300$. Время старта (время "разгона" после получения доступа к шине для передачи) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса (при одновременных запросах - в порядке номеров ЭВМ). Операции с файлами и процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.
26. Консистентное и строго консистентные множества контрольных точек. Дайте оценку накладных расходов на синхронную фиксацию строго консистентного множества контрольных точек для сети из 10 ЭВМ с шинной организацией (без аппаратных возможностей широковещания), если накладные расходы на синхронную фиксацию консистентного множества равны T_1 . Время старта (время "разгона" после получения доступа к шине для передачи сообщения) равно 100, время передачи байта равно 1 ($T_s=100, T_b=1$). Доступ к шине ЭВМ получают последовательно в порядке выдачи запроса на передачу (при одновременных запросах - в порядке номеров ЭВМ). Процессорные операции, включая чтение из памяти и запись в память, считаются бесконечно быстрыми.

Итоговая аттестация аспирантов. Итоговая аттестация аспирантов по дисциплине проводится в соответствии с локальным актом ФИЦ ИПМ им. М.В. Келдыша РАН –

Положением о текущей, промежуточной и итоговой аттестации аспирантов ФИЦ ИПМ им. М.В. Келдыша РАН по программам высшего образования – программам подготовки научно-педагогических кадров в аспирантуре и является обязательной.

Итоговая аттестация по дисциплине осуществляется в форме экзамена в период зачетно-экзаменационной сессии в соответствии с Графиком учебного процесса по приказу (распоряжению заместителя директора по научной работе). Обучающийся допускается к экзамену в случае выполнения всех учебных заданий и мероприятий, предусмотренных настоящей программой. В случае наличия учебной задолженности (пропущенных занятий и (или) невыполненных заданий) аспирант отрабатывает пропущенные занятия и выполняет задания.

Оценивание обучающегося на итоговой аттестации осуществляется с использованием нормативных оценок по 4-х бальной системе (5-отлично, 4-хорошо, 3-удовлетворительно, 2-не удовлетворительно).

На экзамен выносятся одна контрольная задача и один контрольный вопрос.

В качестве контрольных задач на итоговой аттестации используются контрольные задачи, применяемые для оценки промежуточного уровня успеваемости аспиранта.

Примерный перечень рекомендуемых контрольных вопросов для оценки итогового уровня успеваемости аспиранта:

1. Этапы развития вычислительной техники и программного обеспечения.
2. Пакетная ОС, ОС разделения времени, ОС реального времени, распределенные и сетевые ОС.
3. Основные компоненты и логические функции ОС. Базовые понятия: ядро, процесс, ресурс, системные вызовы. Структурная организация ОС.
4. Управление процессами. Определение процесса, типы. Жизненный цикл, состояния процесса. Контекст процесса.
5. Планирование процессов. Уровни планирования. Основные цели планирования. Алгоритмы планирования процессов.
6. Взаимодействие процессов. Разделяемые ресурсы. Критические секции. Взаимное исключение. Тупики.
7. Взаимодействие процессов. Способы реализации взаимного исключения: семафоры Дейкстры, механизм событий, обмен сообщениями.
8. Алгоритм Деккера и алгоритм Петерсона для решения задачи взаимного исключения.
9. Классические задачи синхронизации процессов.
10. Синхронизация в распределенных системах: синхронизация времени, выбор координатора.
11. Основы технологии MPI.
12. Основы технологии OpenMP.
13. Понятие файловой системы. Файл. Типы и атрибуты файлов. Логическая организация файла. Операции над файлами и каталогами. Защита файлов.
14. Семантика одновременного доступа к одному файлу нескольких процессов.
15. Распределенные и параллельные файловые системы.
16. Основные функции ОС по управлению памятью. Типы адресов. Понятие виртуальной памяти. Страничное, сегментное, странично-сегментное распределение памяти.
17. Управление оперативной памятью в распределенных вычислительных системах, распределенная общая память (DSM), алгоритмы реализации, модели консистентности.
18. Управление внешними устройствами. Ввод и вывод информации.
19. Отказоустойчивые системы. Алгоритмы фиксации контрольных точек и восстановления после отказа.
20. Отказоустойчивые системы. Протоколы голосования.

Оценивание аспиранта на итоговой аттестации в форме экзамена

Оценка	Требования к знаниям и критерии выставления оценок
Неудовлетворительно	<p>не решена контрольная задача;</p> <p>основное содержание учебного материала не раскрыто;</p> <p>допущены грубые ошибки в определении понятий и при использовании терминологии;</p> <p>не даны ответы на дополнительные вопросы.</p>
Удовлетворительно	<p>не решена контрольная задача;</p> <p>владение основным объемом знаний по дисциплине;</p> <p>проявляются затруднения в самостоятельных ответах, неточные формулировки;</p> <p>в процессе ответов допускаются ошибки по существу вопросов;</p> <p>экзаменуемый владеет только обязательным минимумом базовых концепций.</p>
Хорошо	<p>решена контрольная задача;</p> <p>раскрыто содержание материала, даны корректные определения понятий;</p> <p>самостоятельно и отчасти при наводящих вопросах даются полноценные ответы на вопросы билета и дополнительные вопросы;</p> <p>не всегда выделяется наиболее существенное, но нет серьезных ошибок в ответах.</p>
Отлично	<p>решена контрольная задача;</p> <p>раскрыто содержание материала, даны корректные определения понятий;</p> <p>допускаются незначительные нарушения последовательности изложения;</p> <p>допускаются небольшие неточности при использовании терминов или в логических выводах;</p> <p>при неточностях задаются дополнительные вопросы, на которые получены корректные ответы.</p>

5. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

1. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. - СПб.: Питер, 2015. - 1120 с.: ил. - (Серия «Классика computer science»). ISBN 978-5-496-01395-6.
2. Таненбаум Э., М. ван Стеен. Распределенные системы. Принципы и парадигмы - СПб.: Питер, 2003. - 877 с: ил. - (Серия «Классика computer science»). ISBN 5-272-00053-6.

Дополнительная литература и Интернет-ресурсы

1. Столлингс В. Операционные системы. Внутреннее устройство и принципы проектирования. 4 изд. - М.: Изд. дом «Вильямс», 2002. - 843 с. ISBN 5-8459-03 10-6.

2. Стивенс У. UNIX - взаимодействие процессов. - СПб.: Питер, 2002. - 576 с. ISBN 5-318-00534-9.
3. Таненбаум Э. Архитектура компьютера. 4-е изд. - СПб.: Питер, 2003. – 704 с. (Серия «Классика computer science»). ISBN 5-318-00298-6.
4. Вахалия Ю. Unix изнутри. - СПб.: Питер, 2003. - 844 с. ISBN 5-94723-013-5.
5. Крюков В.А., Бахтин В.А. Конспект лекций по курсу «Распределенные системы». Факультет ВМК. МГУ им. М.В. Ломоносова (<http://sp.cmc.msu.ru/courses/os/>)
6. Машечкин И.В., Терехин А.Н., Томилин А.Н и др. Материалы по курсу «Операционные системы». Факультет ВМК. МГУ им. М.В. Ломоносова (<http://jaffar.cs.msu.ru/mash/os/>)
7. Антонов А.С. Технологии параллельного программирования MPI и OpenMP: Учеб. пособие. Предисл.: В.А.Садовничий. - М.: Издательство Московского университета, 2012.-344 с.-(Серия "Суперкомпьютерное образование"). ISBN 978-5-211-06343-3.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для обеспечения интерактивных методов обучения для чтения лекций требуется аудитория с мультимедиа (возможен вариант с интерактивной доской).

Для проведения дискуссий и круглых столов, возможно, использование аудиторий со специальным расположением столов и стульев.

ИСПОЛНИТЕЛИ (разработчики программы):

Бахтин В.А., ФИЦ ИПМ им. М.В. Келдыша РАН, заведующий сектором, к.ф.-м.н.