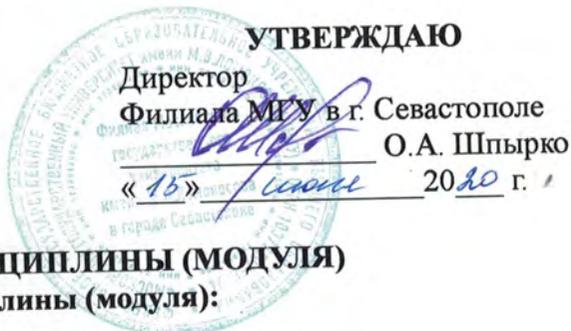


Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования  
Московский государственный университет имени М.В. Ломоносова  
филиал МГУ в г. Севастополе  
факультет компьютерной математики

**УТВЕРЖДЕНО** кафедра программирования  
на 20 21 - 20 22 учебный год  
Методическим советом Филиала

Протокол № 8 от «31» 08 2021 г.

Заместитель директора по учебной работе  
Заведующий кафедрой



**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**  
Наименование дисциплины (модуля):

*РАСПРЕДЕЛЕННЫЕ СИСТЕМЫ*

*код и наименование дисциплины (модуля)*

**Уровень высшего образования:**  
*бакалавриат*

**Направление подготовки:**

*01.03.02 Прикладная математика и информатика*

*(код и название направления/специальности)*

**Направленность (профиль) ОПОП:**  
*общий*

*(если дисциплина (модуль) относится к вариативной части программы)*

**Форма обучения**

*очная*

Рабочая программа рассмотрена  
на заседании кафедры программирования  
протокол № 3 от «28» сентября 2020 г.  
Руководитель ОП 01.03.02 «Прикладная  
математика и информатика»  
  
(Н. В. Лактионова)  
(подпись)

Рабочая программа одобрена  
Методическим советом  
Филиала МГУ в г. Севастополе  
Протокол № 6 от «10» июня 2020 г.  
  
(А.В. Мартынкин)  
(подпись)

Рабочая программа дисциплины (модуля) разработана в соответствии с самостоятельно установленным МГУ образовательным стандартом (ОС МГУ) для реализуемых основных профессиональных образовательных программ бакалавриата, магистратуры, реализуемых последовательно по схеме интегрированной подготовки по направлениям 01.03.02, 01.04.02 «Прикладная математика и информатика» в редакции приказа МГУ от 30 декабря 2016 г.

Год (годы) приема на обучение 2016-2018

1. Дисциплина относится к *вариативной части ОПОП ВО*.

2. Входные требования для освоения дисциплины (модуля): учащиеся должны владеть знаниями по операционным системам, введению в сети ЭВМ и суперкомпьютерам и параллельной обработке данных в объеме, соответствующем программе второго и третьего года обучения основных образовательных программ бакалавриата по укрупненным группам направлений и специальностей 01.00.00 «Математика и механика», 02.00.00 «Компьютерные и информационные науки».

3. Результаты обучения по дисциплине (модулю), соотнесенные с требуемыми компетенциями выпускников.

Планируемые результаты обучения по дисциплине (модулю):

**Знать:**

1. Проблемы создания распределенных операционных систем - операционных систем, которые позволяют рассматривать совокупность независимых компьютеров как единый очень мощный компьютер, использование которого не намного сложнее, чем использование персональной ЭВМ.
2. Основные принципы построения РОС (прозрачность, гибкость, надежность, эффективность, масштабируемость).
3. Особенности технологий MPI и OpenMP для разработки программного обеспечения для распределенных систем и мультипроцессоров.

**Уметь:**

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

**Владеть:**

1. Техникой оценки времени работы различных алгоритмов (выбора координатора, прохождения критических секций, надежных и неделимых широковещательных рассылок сообщений, фиксации консистентного и строго консистентного множества контрольных точек), используемых в распределенных системах, при заданных параметрах - количестве ЭВМ и характеристиках сети (времени старта и времени передачи байта).

2. Техникой оценки времени выполнения операции модификации/чтения переменной из распределенной разделяемой памяти (DSM) для заданной модели консистентности памяти и при заданных параметрах сети (времени старта и времени передачи байта).

4. Формат обучения: лекции проводятся в традиционной манере с использованием компьютера и мультимедийного проектора для демонстрации слайдов.

5. Объем дисциплины (модуля) составляет 3 з.е., в том числе 36 академических часов, отведенных на контактную работу обучающихся с преподавателем, 72 академических часа на самостоятельную работу обучающихся.

6. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий.

Наименование и краткое содержание разделов и тем дисциплины (модуля),  Форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	В том числе			
		Контактная работа (работа во взаимодействии с преподавателем) Виды контактной работы, часы			Самостоятельная работа обучающегося, часы <i>(виды самостоятельной работы – эссе, реферат, контрольная работа и пр. – указываются при необходимости)</i>
		Занятия лекционного типа*	Занятия семинарского типа*	Всего	
1. Введение в параллельные и распределенные системы	8	4	0	4	4
2. Операционные системы многопроцессорных ЭВМ с общей памятью	12	6	0	6	6

3. Коммуникации в распределенных системах	<b>12</b>	6	0	<b>6</b>	<b>6</b>
4. Синхронизация в распределенных системах	<b>12</b>	6	0	<b>6</b>	<b>6</b>
5. Распределенные и параллельные файловые системы	<b>8</b>	4	0	<b>4</b>	<b>4</b>
6. Распределенная разделяемая память (DSM)	<b>12</b>	6	0	<b>6</b>	<b>6</b>
7. Обеспечение надежности	<b>8</b>	4	0	<b>4</b>	<b>4</b>
Промежуточная аттестация: устный экзамен	<b>36</b>	0	0	<b>0</b>	<b>36</b>
<b>Итого</b>	<b>108</b>	<b>36</b>	<b>0</b>	<b>36</b>	<b>72</b>

7. Фонд оценочных средств (ФОС) для оценивания результатов обучения по дисциплине (модулю)

7.1. Типовые контрольные задания или иные материалы для проведения текущего контроля успеваемости.

Текущий контроль успеваемости не предусмотрен рабочей программой.

7.2. Типовые контрольные задания или иные материалы для проведения промежуточной аттестации.

Вопросы к экзамену.

Какие аппаратные механизмы необходимы для организации мультипрограммного режима? Как обеспечить мультипрограммный режим без этих механизмов? Как обеспечить, если отсутствует только один из этих механизмов?

2. Имеется механизм двоичных семафоров. Опираясь на него, реализуйте P-операцию и V-операцию для общего (читающего) семафора. Активное ожидание освобождения семафора не допускается.

3. 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.

4. 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 ];
semaphore s[ L1 ][ L2 ]; /* массив двоичных семафоров с нулевым начальным значением */
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 ]);
    }
```

5. 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).

6. Какие модели консистентности памяти удовлетворяют алгоритму Деккера (алгоритм без каких-либо изменений будет работать правильно), а какие нет? Объясните ответ.

7. Какие модели консистентности памяти удовлетворяют алгоритму Петерсона (алгоритм без каких-либо изменений будет работать правильно), а какие нет? Объясните ответ.

8. В транспьютерной матрице размером  $4 \times 4$ , 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.

9. В транспьютерной матрице размером  $4 \times 4$ , 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.

10. В транспьютерной матрице размером  $4 \times 4$ , 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.

11. В транспьютерной матрице размером  $4 \times 4$ , 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 редукции одновременно. А сколько времени потребуется для нахождения максимума среди 64 чисел в матрице  $8 \times 8$ ? 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.

12. В транспьютерной матрице размером  $4 \times 4$ , 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 буферизуемом 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.

13. В транспьютерной матрице размером  $4 \times 4$ , 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.

14. 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.

15. 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.

16. 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.

17. 15 процессов, находящихся в узлах транспьютерной матрицы размером  $4 \times 4$ , 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.

18. 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.

19. Сколько времени потребует выбор координатора среди 16 процессов, находящихся в узлах транспьютерной матрицы размером  $4 \times 4$ , 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.

20. Какие принципиальные решения приходится принимать при обеспечении файлового сервиса?

21. Интерфейс сервера директорий.

22. Семантика разделения файлов.
23. Серверы с состоянием и без состояния. Достоинства и недостатки.
24. Алгоритмы обеспечения консистентности кэшей в распределенных файловых системах.
25. Способы организации размножения файлов и коррекции копий.
26. 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 широковещания) 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.
27. 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 широковещания), 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.
28. 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 широковещания), 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.
29. 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 широковещания), 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.
30. 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 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.

31. 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.

32. 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.

33. 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.

34. Протоколы голосования. Алгоритмы и применение. 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.

35. 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.

<b>ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ результатов обучения (РО) по дисциплине (модулю)</b>				
Оценка	2	3	4	5
РО и соответствующие виды оценочных средств				
<b>Знания</b> <i>Экзамен</i>	Отсутствие знаний	Фрагментарные знания	Общие, но не структурированные знания	Сформированные систематические знания
<b>Умения</b> <i>Экзамен</i>	Отсутствие умений	В целом успешное, но не систематическое умение	В целом успешное, но содержащее отдельные пробелы умение (допускает неточности принципиального характера)	Успешное и систематическое умение
<b>Навыки (владения, опыт деятельности)</b> <i>Экзамен</i>	Отсутствие навыков (владений, опыта)	Наличие отдельных навыков (наличие фрагментарного опыта)	В целом, сформированные навыки (владения), но используемые не в активной форме	Сформированные навыки (владения), применяемые при решении задач

<b>Соответствие результатов обучения и компетенций, в развитии которых участвует дисциплина (модуль)</b>	
Результаты обучения	Компетенция, с частичным формированием которой связано достижение результата обучения
<b>Знать:</b>	

<ol style="list-style-type: none"> <li>1. Проблемы создания распределенных операционных систем - операционных систем, которые позволяют рассматривать совокупность независимых компьютеров как единый очень мощный компьютер, использование которого не намного сложнее, чем использование персональной ЭВМ.</li> <li>2. Основные принципы построения РОС (прозрачность, гибкость, надежность, эффективность, масштабируемость).</li> <li>3. Особенности технологий MPI и OpenMP для разработки программного обеспечения для распределенных систем и мультипроцессоров.</li> </ol> <p><b>Владеть:</b></p> <ol style="list-style-type: none"> <li>1. Техникой оценки времени работы различных алгоритмов (выбора координатора, прохождения критических секций, надежных и неделимых широковещательных рассылок сообщений, фиксации консистентного и строго консистентного множества контрольных точек), используемых в распределенных системах, при заданных параметрах - количестве ЭВМ и характеристиках сети (времени старта и времени передачи байта).</li> <li>2. Техникой оценки времени выполнения операции модификации/чтения переменной из распределенной разделяемой памяти (DSM) для заданной модели консистентности памяти и при заданных параметрах сети (времени старта и времени передачи байта).</li> </ol>	ОПК-3.Б
<p><b>Уметь:</b></p> <ol style="list-style-type: none"> <li>1. Использовать различные механизмы и алгоритмы для синхронизации процессов/нитей при доступе к разделяемой памяти, к общей файловой системе.</li> <li>2. Реализовать коллективные MPI-операции рассылки/сбора данных при помощи сообщений точка-точка.</li> </ol>	ПК-4.Б

#### 8. Ресурсное обеспечение:

##### Основная литература:

1. Крюков В.А. Распределенные операционные системы. <http://sp.cs.msu.su> в разделе "Информация".

2. Таненбаум Э., Стен М. ван. Распределенные системы. Принципы и парадигмы. – СПб.: Питер, 2003. (ISBN 5272000536, страниц: 877).
3. Таненбаум Э. Современные операционные системы. 3-е изд. – СПб.: Питер, 2010. (ISBN 9785498073064, страниц: 1120).

Информационные справочные системы:

1. MPI: A Message-Passing Interface Standard. [HTML] (<http://www.mpi-forum.org>).
2. OpenMP Application Program Interface. Version 3.1. July 2011. [HTML] (<http://www.openmp.org/mp-documents/OpenMP3.1.pdf>).

Материально-техническое обеспечение:

аудитория с партами, компьютером и мультимедийным проектором для демонстрации слайдов

9. Язык преподавания - русский.

10. Преподаватели:

профессор факультета ВМК МГУ В.А. Крюков,  
доцент факультета ВМК МГУ В.А. Бахтин.

11. Авторы программы:

профессор факультета ВМК МГУ В.А. Крюков,  
доцент факультета ВМК МГУ В.А. Бахтин.