МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное**

**образовательное учреждение высшего образования**

**«Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского»**

|  |
| --- |
| Институт информационных технологий, математики и механики |

|  |
| --- |
| УТВЕРЖДАЮ: |

|  |  |  |
| --- | --- | --- |
| Директор |  | В.П. Гергель |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| « |  | » |  |  | 2017 г. |

**Рабочая программа дисциплины**

|  |
| --- |
| Операционные системы |

Уровень высшего образования

|  |
| --- |
| **Бакалавриат** |

Направление подготовки

|  |
| --- |
| **01.03.02 Прикладная математика и информатика** |

Направленность образовательной программы

|  |
| --- |
| **Прикладная математика и информатика (общий профиль)** |

Квалификация

|  |
| --- |
| **Бакалавр** |

Форма обучения

|  |
| --- |
| **очная** |

Нижний Новгород

2017

**1. Место и цели дисциплины в структуре ОПОП**

**Место дисциплины в структуре ОПОП**

Дисциплина (Б1.Б.13) «Операционные системы» относится к базовой части ОПОП бакалавриата по направлению подготовки 01.03.02 «Прикладная математика и информатика». Обязательна для освоения в 4 семестре второго года обучения.

**Цель освоения дисциплины**

Освоение дисциплины «Операционные системы» преследует следующие цели:

– изучение принципов построения и функционирования операционных систем;

– изучение базовых методов и алгоритмов используемых различными подсистемами ОС;

– формирование у слушателей целостного представления об условиях выполнения прикладных программ;

– изучение особенностей работы многопроцессных и многопоточных приложений;

– получение навыков разработки программ для различных операционных сред.

1. **Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников)**

|  |  |
| --- | --- |
| **Формируемые компетенции** | **Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций** |
| *Способность к коммуникации в устной и письменной формах на русском и иностранном языках для решения задач межличностного и межкультурного взаимодействия (ОК-5)**(Начальный этап)* | *УМЕТЬ**У1 (ОК-5) воспринимать, обобщать и анализировать информацию;**У2 (ОК-5) логически верно, аргументированно и ясно строить устную и письменную речь;**ВЛАДЕТЬ**В1 (ОК-5) способностью уточнить, переспросить, задать вопрос на профессиональную тему;**В2 (ОК-5) навыками работы с литературой и другими дополнительными материалами.* |
| *Способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения (ПК-7)**(Базовый этап)* | *ЗНАТЬ**З1 (ПК-7) классификации, характеристики и принципы построения операционных систем;**З2 (ПК-7) понятия процесса и потока, их свойства и операции над ними;**З3 (ПК-7) свойства архитектуры вычислительной системы, существенно используемые ОС;**З4 (ПК-7) понятие планирования и базовые алгоритмы;**З5 (ПК-7) проблему синхронизации и подходы к ее решению;**З6 (ПК-7) основные механизмы межпроцессного взаимодействия;**З7 (ПК-7) принципы функционирования подсистемы управления файлами.**УМЕТЬ**У1 (ПК-7) разрабатывать программы с учетом возможностей и особенностей целевой ОС.**ВЛАДЕТЬ**В1 (ПК-7) навыками разработки программ для различных операционных сред.* |

1. **Структура и содержание дисциплины «Операционные системы»**

Объем дисциплины составляет 2  зачетные единицы, всего 72  часа, из которых  32  часа составляет контактная работа обучающегося с преподавателем:

32  часа занятий лекционного типа.

40  часов на самостоятельную работу обучающегося.

Содержание дисциплины

|  |  |  |
| --- | --- | --- |
| **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), часы**из них | **Самостоятельная работа обучающегося, часы** |
| **Занятия лекционного типа** | **Занятия семинарского типа** | **Занятия лабораторного типа** |  | **Всего** |
| Очная | Очно-заочная | Заочная | Очная | Очно-заочная | Заочная | Очная | Очно-заочная | Заочная | Очная | Очно-заочная | Заочная |  |  |  | Очная | Очно-заочная | Заочная | Очная | Очно-заочная | Заочная |
| Основные понятия и определения теории операционных систем.Цели и задачи курса. История появления и направления эволюции ОС. Функции ОС. Классификации ОС (по назначению, по режиму обработки задач, по особенностям архитектуры, по способам взаимодействия с пользователем). Дополнительные критерии оценки ОС. Недетализированные примеры архитектур (Windows семейства NT, UNIX).Ресурс, классификация ресурсов. Операционная среда (прикладная среда). Процесс. Поток. Диаграмма состояний потока. Создание и завершение процесса. Создание и завершение потока. | 12 |  |  | 6 |  |  |  |  |  |  |  |  |  |  |  | 6 |  |  | 6 |  |  |
| Недетализованные модели объектов аппаратного уровняСпособы адресации памяти и соответствующие им управляющие структуры; линейная; сегментная; страничная; сегментно-страничная. Многозадачный режим работы процессора. Контекст задачи; переключение задач; уровни привилегий; передача управления между уровнями привилегий. Прерывания и их обработка. | 10 |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  | 4 |  |  | 6 |  |  |
| Управление ресурсом «Центральный процессор»Виды планирования. Критерии оценки алгоритмов планирования. Вытесняющие и невытесняющие алгоритмы. Алгоритмы, использующие приоритеты. Алгоритмы краткосрочного планирования FIFO, SJN, SRT, RR, MLFQ. Алгоритмы планирования Windows и UNIX/Linux. | 8 |  |  | 2 |  |  |  |  |  |  |  |  |  |  |  | 2 |  |  | 6 |  |  |
| Синхронизация выполнения потоков/процессовКритические ресурсы и критические секции. Задача взаимного исключения. Использование запрета прерываний. Алгоритм Деккера (Петерсона), алгоритм булочной. Активное ожидание: использование операций "проверка и установка", "обмен". Семафоры. Мьютексы. Мониторы, условные переменные. Задачи "поставщик-потребитель", "читатели-писатели", "Обедающие философы". Взаимоблокировка (тупик). Необходимые условия возникновения тупика. Предотвращение тупиков. Избегание тупиков. Алгоритм банкира. Граф процесс-ресурс. Редукция. Устранение тупиков. | 14 |  |  | 8 |  |  |  |  |  |  |  |  |  |  |  | 8 |  |  | 6 |  |  |
| Передача данных между потоками/процессами.Типы механизмов передачи. Разделяемая память. Потоковая передача. Очереди сообщений. | 7 |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  | 6 |  |  |
| Управление ресурсом «оперативная память»Простое непрерывное распределение памяти. Схема с разделами фиксированного размера. Схема с разделами переменного размера. Использование ВАП на основе страничного преобразования. Стратегии выборки, размещения, замещения. Алгоритмы замещения областей памяти: Biledy, Random, FIFO, LRU, NFU, Second chance, Clock. Внешняя и внутренняя фрагментация, перемещаемость программ, оверлеи, рабочее множество, своппинг. | 4 |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  | 4 |  |  | – |  |  |
| Примеры механизмов межпоточного (межпроцессного) взаимодействия в UNIX и Win32/Win64.Объекты синхронизации в Windows. Реализации семафоров, мьютексов, очередей сообщений, разделяемой памяти в Windows. Реализации неименованных и именованных каналов, сигналов, семафоров, мьютексов, очередей сообщений, разделяемой памяти в UNIX. | 5 |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  | 4 |  |  |
| Долгосрочное хранение данныхУровни архитектуры подсистемы управления файлами. Типы объектов файловых систем в UNIX. Структура адресного пространства устройств долговременного хранения, временные характеристики устройств. Структура UNIX FS. Атрибуты суперблока. Атрибуты i-node. Хранение информации о размещении данных файла в UNIX FS. Структура каталога в UNIX FS. UNIX Fast File System (FFS). Journalled FS, Log-Structured FS. Структуры, используемые ядром для обеспечения доступа процессов к файлам. Атрибуты записей. Операции open() и close(). | 12 |  |  | 6 |  |  |  |  |  |  |  |  |  |  |  | 6 |  |  | 6 |  |  |
| В т.ч. текущий контроль | 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Промежуточная аттестация - зачет  |

1. **Образовательные технологии**

При обучении данной дисциплине используются следующие формы занятий: лекционные (проводятся в терминал-классе).

Лекции читаются для нескольких групп (потока обучения) одновременно. Лекционное время используется для изучения базовых концепций и алгоритмов, используемых в операционных системах, а также обсуждению специфики их реализации в существующих распространенных ОС.

1. **Учебно-методическое обеспечение самостоятельной работы обучающихся**
	1. **Виды самостоятельной работы студентов**

Предполагаются следующие виды самостоятельной работы студентов:

– Изучение учебной литературы (см. перечень образовательных материалов).

– Решение учебных задач (использование различных механизмов ОС семейств Windows и UNIX). Основное внимание уделяется рассмотрению множества доступных прикладному программисту ОС-специфичных механизмов и развитию у студентов навыков их использования в наиболее распространенных семействах ОС – Windows и UNIX (используется ОС Linux).

* 1. **Образовательные материалы для самостоятельной работы студентов**
1. Материалы курса лекций «Операционные системы»
(<http://e-learning.unn.ru/enrol/index.php?id=255>).
2. Мартемьянов, Ю.Ф. Операционные системы. Концепции построения и обеспечения безопасности [Электронный ресурс] : учеб. пособие / Ю.Ф. Мартемьянов, А.В. Яковлев, А.В. Яковлев. — Электрон. дан. — Москва: Горячая линия-Телеком, 2011. — 332 с. — Режим доступа: https://e.lanbook.com/book/5176.
3. **Фонд оценочных средств для промежуточной аттестации по дисциплине**, включающий:
	1. Перечень компетенций выпускников образовательной программы с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования

*Оценка уровня формирования компетенции (ПК-7)*

|  |  |  |
| --- | --- | --- |
| Индикаторы компетенции | Критерии оценивания (дескрипторы) | Шкала оценивания |
| ***ЗНАТЬ****З1 (ПК-7) классификации, характеристики и принципы построения операционных систем;**З2 (ПК-7) понятия процесса и потока, их свойства и операции над ними;**З3 (ПК-7) свойства архитектуры вычислительной системы, существенно используемые ОС;**З4 (ПК-7) понятие планирования и базовые алгоритмы;**З5 (ПК-7) проблему синхронизации и подходы к ее решению;**З6 (ПК-7) основные механизмы межпроцессного взаимодействия;**З7 (ПК-7) принципы функционирования подсистемы управления файлами.****УМЕТЬ****У1 (ПК-7) разрабатывать программы с учетом возможностей и особенностей целевой ОС.****ВЛАДЕТЬ****В1 (ПК-7) навыками разработки программ для различных операционных сред.* | Отсутствие знаний материала, отсутствие способности решения стандартных задач, полное отсутствие навыков, предусмотренных компетенцией. | Плохой уровеньформирования компетенции.«Плохо». |
| Наличие грубых ошибок в основном материале, наличие грубых ошибок при решении стандартных задач, отсутствие навыков, предусмотренных данной компетенцией | Неудовлетворительный уровень формирования компетенции.«Неудовлетворительно». |
| **Знать** некоторые основные понятия, изучаемые в рамках дисциплины (З1–З7). **Уметь** У1 с погрешностями. **Владеть** некоторыми основными навыками, демонстрируя их в стандартных ситуациях. | Удовлетворительный уровень формирования компетенции.«Удовлетворительно». |
| **Знать** большинство основных понятий, изучаемых в рамках дисциплины (З1–З7).**Уметь** У1 с незначительными погрешностями. **Владеть** основными навыками, демонстрируя их в стандартных ситуациях | Хороший уровеньформирования компетенции.«Хорошо». |
| **Знать** основные понятия, изучаемые в рамках дисциплины (З1–З7). **Уметь** У1 с незначительными погрешностями. **Владеть** всеми основными навыками, демонстрируя их в стандартных ситуациях, в том числе при решении дополнительных задач. | Очень хороший уровеньформирования компетенции.«Очень хорошо». |
| **Знать** основной материал, предусмотренный компетенцией, без ошибок и погрешностей. **Уметь** У1 в полном объеме. **Владеть** всеми навыками, демонстрируя их не только в стандартных ситуациях, но и при решении нестандартных задач. | Отличный уровеньформирования компетенции.«Отлично». |
| **Знать** основной и дополнительный материал, предусмотренный компетенцией, без ошибок и погрешностей. **Уметь** У1 в полном объеме. **Владеть** всеми навыками, демонстрируя их не только в стандартных ситуациях, но и при решении нестандартных задач. | Превосходный уровеньформирования компетенции.«Превосходно». |

*Оценка уровня формирования компетенции (ОК-5)*

|  |  |  |
| --- | --- | --- |
| Индикаторы компетенции | Критерии оценивания (дескрипторы) | Шкала оценивания |
| ***УМЕТЬ****У1 (ОК-5) воспринимать, обобщать и анализировать информацию;**У2 (ОК-5) логически верно, аргументированно и ясно строить устную и письменную речь;****ВЛАДЕТЬ****В1 (ОК-5) способностью уточнить, переспросить, задать вопрос на профессиональную тему;**В2 (ОК-5) навыками работы с литературой и другими дополнительными материалами.* | Соответствующие личностные качества не сформированы. | Плохой уровеньформирования компетенции.«Плохо». |
| Уровень формирования личностных качеств недостаточен для достижения основных целей обучения. | Неудовлетворительный уровень формирования компетенции.«Неудовлетворительно». |
| Личностные качества сформированы на уровне, минимально необходимом для достижения основных целей обучения. | Удовлетворительный уровень формирования компетенции.«Удовлетворительно». |
| Личностные качества в целом сформированы. | Хороший уровеньформирования компетенции.«Хорошо». |
| Уровень формирования личностных качеств достаточен для достижения основных целей обучения. | Очень хороший уровеньформирования компетенции.«Очень хорошо». |
| Личностные качества сформированы на высоком уровне. | Отличный уровеньформирования компетенции.«Отлично». |
| Уровень формирования личностных качеств выше обязательных требований. | Превосходный уровеньформирования компетенции.«Превосходно». |

* 1. Описание шкал оценивания

Промежуточная аттестация по дисциплине проводится в виде зачета.

По результатам итоговой аттестации проставляются оценки «Зачтено» (соответствует уровням оценки компетенций «удовлетворительно» и выше) и «Не зачтено» (соответствует уровням оценки компетенций «плохо» и «неудовлетворительно»).

Для оценивания результатов учебной деятельности студентов при изучении дисциплины используется комбинированная система оценивания, цель которой состоит в следующем:

– объективно оценивать знания студентов;

– дополнительно мотивировать студентов изучать необходимый материал в течение семестра.

Результаты работы студентов оцениваются непрерывно в ходе семестра. При этом учитываются следующие факторы.

– Текущая успеваемость. Необходимым условием выставления оценки «Удовлетворительно» является успешная сдача студентами практических заданий по всем пройденным темам (допускается несдача одной темы). Общее число тем – порядка 15, содержание заданий варьируется в соответствии с составом прочитанного материала.

– Результаты выполнения практических заданий. Большинство из них (проводятся в терминал-классе) предполагают самостоятельное выполнение заданий. Все выполненные задания оцениваются согласно набору критериев ((а) возможность скомпилировать и запустить на выполнение подготовленное решение, (б) корректность алгоритма решения задачи, (в) корректность использования необходимых ОС-специфичных функций, (г) общее качество кода).

В ходе итоговой аттестации студенты должны отчитаться по всем темам, не сданным в процессе контроля текущей успеваемости. Результаты выполнения практических заданий влияют на количество и состав вопросов по темам. По результатам итоговой аттестации проставляются оценки согласно критериям оценки компетенций.

* 1. Критерии и процедуры оценивания результатов обучения по дисциплине (модулю), характеризующих этапы формирования компетенций

Для оценивания результатов обучения в виде знаний используются следующие процедуры и технологии:

– письменные ответы на вопросы.

Для оценивания результатов обучения в виде умений и владений используются следующие процедуры и технологии:

– практические задания, включающие постановку одной учебной задачи низкой или средней сложности в виде краткого описания результата, который нужно получить, и перечисления алгоритмов и механизмов, которые следует использовать.

* 1. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций и (или) для итогового контроля сформированности компетенции.

Примеры контрольных вопросов, используемых при проведении текущего (контрольные работы) и итогового (зачет) контроля успеваемости для оценивания результатов формирования компетенций ОК-5, ПК-7.

1. Недетализированные примеры архитектур - Windows NT

2. Недетализированные примеры архитектур – UNIX.

3. Процесс. Поток. Диаграмма состояний последовательного исполнения потока.

4. Структуры данных ОС, связанные с процессом. Создание процесса. Завершение процесса.

5. Структуры данных ОС, связанные с потоком. Создание потока. Завершение потока.

6. Сегментная организация памяти.

7. Организация ВАП процесса на основе страничного преобразования.

8. Многозадачный режим работы процессора. Уровни привилегий (кольца защиты).

9. Контекст задачи. Переключение задач. Передача управления между уровнями привилегий.

10. Прерывания. Обработка прерываний.

11. Критерии сравнения алгоритмов планирования.

12. Алгоритмы планирования FIFO, SJN, SRT.

13. Краткосрочное планирование. RR, алгоритмы, использующие приоритеты.

14. Критические ресурсы и критические секции процессов. Постановка задачи взаимного исключения.

15. Алгоритм Петерсона. Алгоритм булочной.

16. Использование операции “проверка и установка”. Семафорные примитивы Дийкстры.

17. Постановка и решение задачи “Читатели-писатели”.

18. Постановка и решение задачи "Производители-потребители".

19. Постановка и решение задачи "Обедающие философы".

20. Простое непрерывное распределение памяти. Случаи одной и нескольких исполняемых задач.

21. Алгоритмы замещения областей памяти (Оптимальный, FIFO, LRU, LFU).

22. Файлы, проецируемые в память.

23. Сигналы UNIX. Очереди сообщений UNIX. Именованные каналы UNIX.

24. Иерархия памяти, характеристики устройств долговременного хранения данных.

25. Традиционная файловая система UNIX, Berkeley FFS.

26. Журналируемая файловая система (JFS). Файловая система с журнальной структурой (LSFS).

Примеры практических заданий для самостоятельной работы для проведения текущего контроля успеваемости, выставления оценки для оценивания результатов формирования компетенций ОК-5, ПК-7.

СР1. Процессы и операции над ними – 1

Требуется создать программу для ОС UNIX, выполняющую следующие действия:

1. Создание процесса-потомка

Далее действия программы в процессе-родителе и процессе-потомке описываются отдельно.

Действия программы в процессе-родителе

2. Вывести сообщение о том, что программа выполняется в родителе.

3. Дождаться завершения процесса-потомка.

4. Вывести сообщение о завершении процесса-потомка и его код завершения.

Действия программы в процессе-потомке

2. Вывести сообщение о том, что программа выполняется в процессе-потомке.

3. Запустить на выполнение следующую командную строку:

 /bin/ls -la /tmp

СР 2. Процессы и операции над ними – 2

Требуется создать программу для ОС Windows, которая должна выполнять следующие действия:

1. Создание процесса-потомка, в котором выполняется программа "Notepad", запущенная с агрументом вызова "C:\Windows\system.ini".

2. Вывод идентификатора дочернего процесса.

3. Ожидание 10 секунд и завершение дочернего процесса.

СР 3. Потоки и операции над ними

Требуется создать программу, которая должна выполнять следующие действия:

1. Вывод начального значения целочисленной переменной Var, объявленной с квалификатором volatile.

2. Запуск параллельного выполнения 20 потоков, каждый из которых выполняет 100 000 000 изменений переменной Var, адрес которой должен быть передан ему в качестве входного параметра.

При этом 10 потоков 100 000 000 раз увеличивают переменную Var на 1, 10 потоков 100 000 000 раз уменьшают переменную Var на 1.

3. Вывод конечного значения целочисленной переменной Var.

СР 4. Именованные и неименованные каналы

Требуется создать программу, выполняющую запуск следующего конвейера:

ls -la /dev | more

СР 5. Файлы, отображаемый в память

Требуется создать программу, выполняющую следующие действия:

1. Отображение файла в память (имя файла передается первым аргументом командной строки).

2. Изменение порядка следования байт в отображенном блоке памяти на противоположный.

3. Отключение отображения файла в память.

СР 6. Задача Читатели-Писатели

Требуется создать программу, выполняющую следующие действия.

1. Инициализация глобальной переменной, представляющей критические данные.

int Data=0;

2. Создание 5 потоков-писателей и 5 потоков-читателей.

3. Ожидание завершения потоков.

Схема выполнения потока-писателя следующая.

Writer(){

 while( 5 итераций ){

 sleep( rand(от 10 до 20) );

 Вход\_в\_критическую\_секцию

 printf("Writer N %d starts writing. ReadCount=%d, Data=%d\n", номер\_писателя, ReadCount, Data);

 sleep(3);

 printf("Writer N %d ends writing. ReadCount=%d, Data=%d\n", номер\_писателя, ReadCount, Data);

 Data ++;

 Выход\_из\_критической\_секции

 }

}

Схема работы потока-читателя аналогичная за исключением строки Data ++;

(читатель не изменяет данные).

СР 7. Алгоритм банкира

Требуется многократно решить задачу выделения ресурса процессу согласно алгоритму банкира.

6.5.Методические материалы, определяющие процедуры оценивания

Положение о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся в ННГУ от 13.02.2014.

<http://www.unn.ru/pages/general/norm-acts/attest_stud%202014.pdf>

Положение о фонде оценочных средств, утвержденное приказом ректора ННГУ от 10.06.2015 №247-ОД.

1. **Учебно-методическое и информационное обеспечение дисциплины**

а) основная литература:

1. Материалы курса лекций «Операционные системы»
(<http://e-learning.unn.ru/enrol/index.php?id=255>).
2. Мартемьянов, Ю.Ф. Операционные системы. Концепции построения и обеспечения безопасности [Электронный ресурс]: учеб. пособие / Ю.Ф. Мартемьянов, А.В. Яковлев, А.В. Яковлев. — Электрон. дан. — Москва: Горячая линия-Телеком, 2011. — 332 с. — Режим доступа: <https://e.lanbook.com/book/5176>

б) дополнительная литература:

1. Гордеев А. В. - Операционные системы: учеб. для студентов вузов, обучающихся по направлению подготовки бакалавров и магистров "Информатика и вычислительная техника" ... - СПб. [и др.]: Питер, 2004. - 416 с. (20 экз.)
2. В.Карпов, К.Коньков. Основы операционных систем. <http://www.intuit.ru/studies/courses/2192/31/info>
3. В.Карпов, К.Коньков. Основы операционных систем. Практикум. <http://www.intuit.ru/studies/courses/2249/52/info>

в) программное обеспечение и Интернет-ресурсы

1. Gorman M. Understanding The Linux Virtual Memory Manager. 2003 (Электронный источник –<https://www.kernel.org/doc/gorman/>)
2. Rusling D. The Linux Kernel (Электронный источник – <https://www.tldp.org/LDP/tlk/tlk.html>)
3. Marshall A.D. Programming in C UNIX System Calls and Subroutines using C. 1999. (Электронный источник – <http://www.cs.cf.ac.uk/Dave/C/CE.html> )
4. Cross-Referencing Linux (Электронный источник – <http://lxr.linux.no> )
5. The Linux Kernel Archives (Электронный источник – [www.kernel.org](http://www.kernel.org) )
6. **Материально-техническое обеспечение дисциплины**

Имеются в наличии учебные аудитории для проведения занятий лекционного типа, промежуточной аттестации, а также помещения для самостоятельной работы, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет». Наличие рекомендованной литературы.

Для выполнения практических заданий в рамках самостоятельной работы имеется компьютерный класс ПЭВМ с микропроцессором не ниже Pentium IV, объемом памяти не менее 1 ГБ, свободным местом на жестком диске не менее 5 Гб.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 01.03.02 «Прикладная математика и информатика»

Авторы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_С.Н. Карпенко

 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Линев

Рецензент: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Заведующий кафедрой МОСТ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.П. Гергель

Программа одобрена методической комиссией Института информационных технологий, математики и механики ННГУ им. Н.И. Лобачевского

от 29 августа 2017года, протокол № 20.