

**Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Московский физико-технический институт  
(национальный исследовательский университет)»**

**УТВЕРЖДЕНО**

**Директор физтех-школы  
электроники, фотоники и  
молекулярной физики**

**В.В. Иванов**

**Рабочая программа дисциплины (модуля)**

<b>по дисциплине:</b>	Теоретические и технические основы численного анализа
<b>по направлению:</b>	Прикладные математика и физика
<b>профиль подготовки:</b>	Физика перспективных технологий: альтернативная энергетика, научное программирование и функциональные материалы Физтех-школа Электроники, Фотоники и Молекулярной Физики кафедра физики высокотемпературных процессов
<b>курс:</b>	4
<b>квалификация:</b>	бакалавр

Семестр, формы промежуточной аттестации: 8 (весенний) - Экзамен

Аудиторных часов: 30 всего, в том числе:

лекции: 30 час.

семинары: 0 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 75 час.

Подготовка к экзамену: 30 час.

Всего часов: 135, всего зач. ед.: 3

Программу составил: П.Р. Левашов, канд. физ.-мат. наук, старший научный сотрудник

Программа обсуждена на заседании кафедры физики высокотемпературных процессов 29.05.2020

## Аннотация

Курс "Теоретические и технические основы численного анализа" предусматривает изучение основ работы с операционной системой UNIX, программирования и решения на компьютерах различных задач вычислительной физики, а также применение полученных знаний и навыков на практике.

Задачи курса:

- изучение основ операционной системы UNIX;
- формирование представлений о языках программирования, изучение особенностей языка программирования C;
- редактирование, компиляция и линковка программ в операционной системе UNIX;
- разработка алгоритма, написание программы, отладка и запуск программ для различных задач вычислительной физики;
- изучение основ параллельного программирования.

По результатам освоения курса студент должен:

Знать:

основы устройства современной компьютерной техники и принципы работы современных операционных систем;

основные принципы работы и основные команды операционной системы UNIX;

классификацию языков программирования, требования к языкам программирования для их использования при моделировании задач вычислительной физики;

принципы структурного программирования;

основы процесса компиляции и линковки программ;

классификацию многопроцессорных вычислительных комплексов;

основные принципы создания параллельных программ для многопроцессорных вычислительных комплексов с распределенной памятью;

основные принципы создания параллельных программ для многопроцессорных вычислительных комплексов с общей памятью.

Уметь:

работать с файловой системой операционной системы UNIX в терминальном режиме;

редактировать файлы программ с помощью стандартных редакторов операционной системы UNIX;

уметь пересылать файлы между компьютерами различными способами;

программировать на языках C и/или FORTRAN;

компилировать и линковать программы, написанные с привлечением стандартных библиотек;

создавать собственные библиотеки подпрограмм и использовать их для написания и компиляции собственных программ;

использовать библиотеку GSL для решения стандартных задач вычислительной математики;

разрабатывать алгоритмы в соответствии с принципами структурного программирования;

разрабатывать простые параллельные алгоритмы, создавать простые параллельные программы для компьютеров с общей и распределенной памятью;

запускать параллельные программы на многопроцессорных вычислительных комплексах;

эффективно использовать информационные технологии и компьютерную технику для достижения необходимых теоретических и экспериментальных результатов.

Владеть:

навыками освоения большого объема информации;

навыками самостоятельной работы в лаборатории и Интернете;

культурой постановки и моделирования физических задач;

навыками грамотной обработки результатов экспериментов и сопоставления с теоретическими и литературными данными;

практикой исследования и решения теоретических и прикладных задач;

практикой работы в операционной системе UNIX, включая написание, редактирование, отладку и запуск последовательных и параллельных программ.

Основное содержание курса изложено в следующих разделах:

1. Компиляция программ в ОС UNIX
2. Методы расчета твердой фазы

## 1. Цели и задачи

### Цель дисциплины

- изучение основ работы с операционной системой UNIX, программирования и решения на компьютерах различных задач вычислительной физики, а также применение полученных знаний и навыков на практике.

### Задачи дисциплины

- изучение основ операционной системы UNIX;
- формирование представлений о языках программирования, изучение особенностей языка программирования C;
- редактирование, компиляция и линковка программ в операционной системе UNIX;
- разработка алгоритма, написание программы, отладка и запуск программ для различных задач вычислительной физики;
- изучение основ параллельного программирования.

## 2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
ПК-1 Способен планировать и проводить научные эксперименты (в избранной предметной области) и (или) теоретические (аналитические и имитационные) исследования	ПК-1.3 Владеет культурой постановки научной задачи и моделирования естественнонаучных объектов и систем
	ПК-1.1 Владеет фундаментальными понятиями, законами и теориями современной физики
	ПК-1.4 Умеет строить математические модели для описания и исследования процессов и явлений в соответствующих научных областях

## 3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны знать:

основы устройства современной компьютерной техники и принципы работы современных операционных систем;  
основные принципы работы и основные команды операционной системы UNIX;  
классификацию языков программирования, требования к языкам программирования для их использования при моделировании задач вычислительной физики;  
принципы структурного программирования;  
основы процесса компиляции и линковки программ;  
классификацию многопроцессорных вычислительных комплексов;  
основные принципы создания параллельных программ для многопроцессорных вычислительных комплексов с распределенной памятью;  
основные принципы создания параллельных программ для многопроцессорных вычислительных комплексов с общей памятью.

уметь:

работать с файловой системой операционной системы UNIX в терминальном режиме;  
редактировать файлы программ с помощью стандартных редакторов операционной системы UNIX;

уметь пересылать файлы между компьютерами различными способами;

программировать на языках С и/или FORTRAN;

компилировать и линковать программы, написанные с привлечением стандартных библиотек;

создавать собственные библиотеки подпрограмм и использовать их для написания и компиляции собственных программ;

использовать библиотеку GSL для решения стандартных задач вычислительной математики;

разрабатывать алгоритмы в соответствии с принципами структурного программирования;

разрабатывать простые параллельные алгоритмы, создавать простые параллельные программы для компьютеров с общей и распределенной памятью;

запускать параллельные программы на многопроцессорных вычислительных комплексах;

эффективно использовать информационные технологии и компьютерную технику для достижения необходимых теоретических и экспериментальных результатов.

владеть:

навыками освоения большого объема информации;

навыками самостоятельной работы в лаборатории и Интернете;

культурой постановки и моделирования физических задач;

навыками грамотной обработки результатов экспериментов и сопоставления с теоретическими и литературными данными;

практикой исследования и решения теоретических и прикладных задач;

практикой работы в операционной системе UNIX, включая написание, редактирование, отладку и запуск последовательных и параллельных программ.

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

##### 4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Компиляция программ в ОС UNIX	3			6
2	Методы расчета твердой фазы	4			7
3	Моделирование случайных процессов	2			7
4	Одномерная и двумерная модели Изинга	2			6
5	Основы ОС UNIX	2			6
6	Параллельное программирование с общей памятью	2			6
7	Параллельное программирование с распределенной памятью	3			6
8	Принципы структурного программирования	3			6
9	Процессы в ОС UNIX	2			7
10	Работа в командной строке ОС UNIX.	2			6
11	Система компиляции программ MAKE	2			6
12	Языки программирования для численного моделирования	3			6

Итого часов	30			75
Подготовка к экзамену	30 час.			
Общая трудоёмкость	135 час., 3 зач.ед.			

#### 4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 8 (Весенний)

##### 1. Компиляция программ в ОС UNIX

Компиляция программ на языке C. Основные этапы компиляции, ключи компилятора gcc на каждом этапе компиляции. Ошибки компиляции на каждом из этапов. Понятие о библиотеках, способы создания и работы с библиотеками.

##### 2. Методы расчета твердой фазы

Двумерная модель Изинга. Понятие о фазовом переходе второго рода. Расчет теплоемкости в двумерном случае, зависимость результатов от параметров моделирования, оценка погрешности моделирования.

##### 3. Моделирование случайных процессов

Моделирование случайных процессов. Понятие о методе Монте-Карло. Алгоритм Метрополиса. Задача о перколяции.

##### 4. Одномерная и двумерная модели Изинга

Одномерная модель Изинга. Аналитические формулы для энергии, магнитного момента и теплоемкости. Моделирование одномерной модели Изинга.

##### 5. Основы ОС UNIX

Основы операционной системы UNIX. История, принципы, влияние на другие операционные системы. Структура операционной системы UNIX.

##### 6. Параллельное программирование с общей памятью

Понятие о программировании с общей памятью, технология OpenMP. Краткое описание современных технологий, которые необходимо освоить для эффективного численного моделирования различных физических задач.

##### 7. Параллельное программирование с распределенной памятью

Понятие о параллельном программировании. Суперкомпьютеры с распределенной памятью. Библиотека MPI, межпроцессорные обмены. Написание, отладка и запуск параллельных задач.

##### 8. Принципы структурного программирования

Принципы структурного программирования. Способы объединения блоков, стандартные блоки. Примеры структурных и неструктурных программ. Отступления от принципов структурного программирования.

##### 9. Процессы в ОС UNIX

Процессы в операционной системе UNIX. Способы управления процессами с помощью сигналов, соответствие некоторых сигналов клавиатурным комбинациям. Понятия о потоках в рамках процесса.

#### 10. Работа в командной строке ОС UNIX.

Основные команды операционной системы UNIX. Работа в командной строке, командный процессор `bash`. Файловая система операционной системы UNIX, работа с файлами и каталогами, права на файлы и каталоги.

#### 11. Система компиляции программ MAKE

Система компиляции программ MAKE. Зависимости между файлами, цели, обработка зависимостей. Команда `make`. Пример компиляции реального проекта.

#### 12. Языки программирования для численного моделирования

Редактирование файлов, основы редактора `vi`. Классификация языков программирования, требования к языкам программирования для написания вычислительных программ. Особенности языков программирования FORTRAN и C.

### 5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Необходимое оборудование для лекций: учебная аудитория, компьютеры и мультимедийное оборудование (проектор), доступ к сети Интернет.

### 6. Перечень рекомендуемой литературы

#### Основная литература

1. Юрий Магда. UNIX. Санкт-Петербург: БХВ-Петербург, 2006.
2. Дейтел Х.М., Дейтел П.Дж. Как программировать на C. М.: Бино, 2009.
3. Хеерман Д.В. Методы компьютерного эксперимента в теоретической физике. М.: Наука, 1990.
4. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. Санкт-Петербург: БХВ-Петербург, 2002.

#### Дополнительная литература

1. Стен Келли-Бутл. Введение в UNIX. М.: «Лори», 1997.
2. Керниган Б., Ритчи Д. Язык программирования C. Санкт-Петербург: Финансы и статистика, 2001.

### 7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Библиотека GSL, <http://www.gnu.org/software/gsl/>

### 8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

Компиляция программ на языке C/C++, <http://www.firststeps.ru/linux/>

### 9. Методические указания для обучающихся по освоению дисциплины (модуля)

Студент, изучающий дисциплину, должен с одной стороны, овладеть общим понятийным аппаратом, а с другой стороны, должен научиться применять теоретические знания на практике. В результате изучения дисциплины студент должен знать основные определения дисциплины, уметь применять полученные знания для решения различных задач.

Успешное освоение курса требует:

- посещения всех занятий, предусмотренных учебным планом по дисциплине;
- ведения конспекта занятий;
- напряжённой самостоятельной работы студента.

Самостоятельная работа включает в себя:

- чтение рекомендованной литературы;
- проработку учебного материала, подготовку ответов на вопросы, предназначенных для самостоятельного изучения;
- решение задач, предлагаемых студентам на занятиях;
- подготовку к выполнению заданий текущей и промежуточной аттестации.

Показателем владения материалом служит умение без конспекта отвечать на вопросы по темам дисциплины.

Важно добиться понимания изучаемого материала, а не механического его запоминания. При затруднении изучения отдельных тем, вопросов, следует обращаться за консультациями к преподавателю.

Возможен промежуточный контроль знаний студентов в виде решения задач в соответствии с тематикой занятий.

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

<b>по направлению:</b>	Прикладные математика и физика
<b>профиль подготовки:</b>	Физика перспективных технологий: альтернативная энергетика, научное программирование и функциональные материалы Физтех-школа Электроники, Фотоники и Молекулярной Физики кафедра физики высокотемпературных процессов
<b>курс:</b>	<u>4</u>
<b>квалификация:</b>	бакалавр
Семестр, формы промежуточной аттестации: 8 (весенний) - Экзамен	
<b>Разработчик:</b>	П.Р. Левашов, канд. физ.-мат. наук, старший научный сотрудник



## 1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
ПК-1 Способен планировать и проводить научные эксперименты (в избранной предметной области) и (или) теоретические (аналитические и имитационные) исследования	ПК-1.3 Владеет культурой постановки научной задачи и моделирования естественнонаучных объектов и систем
	ПК-1.1 Владеет фундаментальными понятиями, законами и теориями современной физики
	ПК-1.4 Умеет строить математические модели для описания и исследования процессов и явлений в соответствующих научных областях

## 2. Показатели оценивания компетенций

В результате изучения дисциплины «Теоретические и технические основы численного анализа» обучающийся должен:

### знать:

- основы устройства современной компьютерной техники и принципы работы современных операционных систем;
- основные принципы работы и основные команды операционной системы UNIX;
- классификацию языков программирования, требования к языкам программирования для их использования при моделировании задач вычислительной физики;
- принципы структурного программирования;
- основы процесса компиляции и линковки программ;
- классификацию многопроцессорных вычислительных комплексов;
- основные принципы создания параллельных программ для многопроцессорных вычислительных комплексов с распределенной памятью;
- основные принципы создания параллельных программ для многопроцессорных вычислительных комплексов с общей памятью.

### уметь:

- работать с файловой системой операционной системы UNIX в терминальном режиме;
- редактировать файлы программ с помощью стандартных редакторов операционной системы UNIX;
- уметь пересылать файлы между компьютерами различными способами;
- программировать на языках C и/или FORTRAN;
- компилировать и линковать программы, написанные с привлечением стандартных библиотек;
- создавать собственные библиотеки подпрограмм и использовать их для написания и компиляции собственных программ;
- использовать библиотеку GSL для решения стандартных задач вычислительной математики;
- разрабатывать алгоритмы в соответствии с принципами структурного программирования;
- разрабатывать простые параллельные алгоритмы, создавать простые параллельные программы для компьютеров с общей и распределенной памятью;
- запускать параллельные программы на многопроцессорных вычислительных комплексах;
- эффективно использовать информационные технологии и компьютерную технику для достижения необходимых теоретических и экспериментальных результатов.

### владеть:

- навыками освоения большого объема информации;
- навыками самостоятельной работы в лаборатории и Интернете;
- культурой постановки и моделирования физических задач;
- навыками грамотной обработки результатов экспериментов и сопоставления с теоретическими и литературными данными;
- практикой исследования и решения теоретических и прикладных задач;
- практикой работы в операционной системе UNIX, включая написание, редактирование, отладку и запуск последовательных и параллельных программ.

## 3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

С целью контроля освоения обучающимися учебного материала проводится устный опрос в начале занятия по теме прошлой лекции или в конце занятия по пройденной теме.

#### **4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся**

Вопросы для подготовки к экзамену:

1. Основные принципы, положенные в основу операционной системы UNIX.
2. Структура операционной системы UNIX.
3. Файловая система операционной системы UNIX.
4. Понятие о процессе, способы управления процессами.
5. Основные этапы компиляции программ на языке C.
6. Ключи компиляции компилятора gcc на этапе препроцессирования, практический пример.
7. Ключи компиляции компилятора gcc на этапе компилирования, практический пример.
8. Ключи компиляции компилятора gcc на этапе линковки, практический пример.
9. Статическая и динамическая линковки. Продемонстрировать на примере.
10. Технология компиляции MAKE. Продемонстрировать на примере.
11. Принципы структурного программирования. Понятие о блоках, способы их комбинирования.
12. Привести примеры алгоритмов, написанных в соответствии/без соответствия с принципами/принципам структурного программирования.
13. Привести примеры, в которых нарушение принципов структурного программирования оправдано. Привести примеры нарушения принципов структурного программирования в языке C.
14. Понятие о датчике случайных чисел, свойства датчиков, критичные для задач численного моделирования.
15. Алгоритмы генерации случайных чисел с заданным распределением.
16. Марковские процессы, алгоритм Метрополиса.
17. Понятие о методе Монте-Карло, метод численного интегрирования с использованием метода Монте-Карло.
18. Понятие о перколяции и бесконечном кластере.
19. Алгоритм нахождения бесконечного кластера на прямоугольной решетке. Порог перколяции в двумерном случае.
20. Задача о случайных блужданиях в одномерном и двумерном случаях.
21. Одномерная модель Изинга, формула для вычисления энергии, магнитного момента и теплоемкости.
22. Алгоритм моделирования ферромагнетика в модели Изинга в каноническом ансамбле с помощью алгоритма Метрополиса.
23. Расчет средней энергии и магнитного момента в одномерной модели Изинга.
24. Расчет теплоемкости в двумерной модели Изинга. Фазовый переход второго рода (ферромагнетик-парамагнетик).
25. Суперкомпьютеры с распределенной памятью. Технология программирования для компьютеров с распределенной памятью, библиотека MPI. Практический пример параллельного алгоритма для задачи численного интегрирования.
26. Суперкомпьютеры с общей памятью. Поток, технология программирования OpenMP. Практический пример параллельного алгоритма для задачи численного интегрирования.
27. Практическая задача: численное решение задачи Коши для обыкновенного дифференциального уравнения с заданной точностью с помощью библиотеки GSL.
28. Практическая задача: вычисление определенного интеграла с помощью библиотеки GSL с заданной точностью.
29. Практическая задача: вычисление несобственного интеграла с помощью библиотеки GSL с заданной точностью.
30. Практическая задача: вычисление суммы ряда с заданной точностью с помощью библиотеки GSL.
31. Практическая задача: интерполяция заданного множества точек функции кубическим сплайном и вычисление производной интерполяционной функции в заданной точке с помощью библиотеки GSL.

32. Практическая задача: нахождение всех корней алгебраического уравнения с помощью библиотеки GSL.

33. Практическая задача: решение системы линейных уравнений с помощью библиотеки GSL.

Примеры экзаменационных билетов.

Пример 1.

1. Привести примеры алгоритмов, написанных в соответствии/без соответствия с принципами/принципам структурного программирования.
2. Практическая задача: интерполяция заданного множества точек функции кубическим сплайном и вычисление производной интерполяционной функции в заданной точке с помощью библиотеки GSL.

Пример 2.

1. Структура операционной системы UNIX.
2. Практическая задача: вычисление определенного интеграла с помощью библиотеки GSL с заданной точностью.

### Критерии оценивания

Оценка отлично 10 баллов - выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины, проявляющему интерес к данной предметной области, продемонстрировавшему умение уверенно и творчески применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка отлично 9 баллов - выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка отлично 8 баллов - выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, правильное обоснование принятых решений, с некоторыми недочетами.

Оценка хорошо 7 баллов - выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но недостаточно грамотно обосновывает полученные результаты.

Оценка хорошо 6 баллов - выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности.

Оценка хорошо 5 баллов - выставляется студенту, если он в основном знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач достаточно большое количество неточностей.

Оценка удовлетворительно 4 бала - выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он освоил основные разделы учебной программы, необходимые для дальнейшего обучения, и может применять полученные знания по образцу в стандартной ситуации.

Оценка удовлетворительно 3 бала - выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, допускающему ошибки в формулировках базовых понятий, нарушения логической последовательности в изложении программного материала, слабо владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и с трудом применяет полученные знания даже в стандартной ситуации.

Оценка неудовлетворительно 2 бала - выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных принципов и не умеет использовать полученные знания при решении типовых задач.

Оценка неудовлетворительно 1 бал - выставляется студенту, который не знает основного содержания учебной программы дисциплины, допускает грубейшие ошибки в формулировках базовых понятий дисциплины и вообще не имеет навыков решения типовых практических задач.

#### **5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности**

При проведении устного экзамена обучающемуся предоставляется 30 минут на подготовку. Опрос обучающегося по билету на устном экзамене не должен превышать одного астрономического часа.