

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

УТВЕРЖДЕНО

**Директор физтех-школы
радиотехники и компьютерных
технологий**

А.В. Дворкович

	Рабочая программа дисциплины (модуля)
по дисциплине:	Программирование систем
по направлению:	Прикладные математика и физика
профиль подготовки:	Радиотехника и компьютерные технологии Физтех-школа Радиотехники и Компьютерных Технологий кафедра электронных вычислительных машин
курс:	4
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 7 (осенний) - Дифференцированный зачет

Аудиторных часов: 15 всего, в том числе:

лекции: 15 час.

семинары: 0 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 30 час.

Всего часов: 45, всего зач. ед.: 1

Программу составили:

Е.Г. Кречетов, канд. техн. наук

В.А. Поздняков, канд. техн. наук

Н.Б. Преображенский, канд. техн. наук, старший научный сотрудник

Программа обсуждена на заседании кафедры электронных вычислительных машин 04.06.2020

Аннотация

В развитие курса «Цифровые методы обработки в системах передачи данных» в данном курсе рассматриваются вопросы реализации программных информационных систем. Обсуждаются приемы эффективной организации работ по разработке программных продуктов. Создание программных систем предполагает наличие определенной методики коллективного труда по проектированию, отладке и комплексированию программных продуктов. В курсе рассматриваются вопросы поэтапной реализации программных проектов, создания программ как объектов обработки и анализа. Отдельное внимание уделяется вопросам прикладного характера решения задач программирования, необходимости соответствия выбранных приемов и алгоритмов особенностям моделей прикладных областей, для которых выполняется программирование.

Слушатели курса ориентируются на структурное осмысление проектных задач, в базисе усвоенных понятий и терминов, и четкое изложение технических вопросов в устной и письменной речи. Контрольные задания формулируются как прикладные проектные задачи средней сложности.

На практических и семинарских занятиях проводятся анализ, обсуждения и сравнительные оценки ряда современных приемов прикладного программирования с целью закрепления усвоенного теоретического материала, а также изучения реализации прикладных задач использования вычислительной техники.

Для успешного прохождения курса необходимо посещение и конспектирование лекций, выполнение практических заданий и самостоятельная работа с дополнительными литературными источниками.

1. Цели и задачи

Цель дисциплины

Дать обучаемым по специальности прикладная математика и физика (специализация –электронные вычислительные машины), дать обучаемым по специальности прикладная математика и физика (специализация –электронные вычислительные машины), комплекс сведений о современных приемах построения программных систем, имеющих различное назначение и реализацию, познакомить со структурой и принципами создания и сопровождения библиотек программ, базовых библиотек и прикладных программных подсистем.

Задачи дисциплины

- приобретение слушателями теоретических знаний и практических умений и навыков в области создания и использования прикладного программного обеспечения;
- раскрытие сущности и значения задач проектирования программных систем, определения теоретических, концептуальных, методологических и организационных основ проектной деятельности;
- формирования системного подхода в сфере проектирования программных систем и привития инженерной культуры, т.е. умения целенаправленно работать с информацией, используя ее для решения профессиональных вопросов;
- подготовка слушателей к изучению смежных дисциплин специализации.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
	УК-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки
	УК-1.5 Определяет и оценивает практические последствия возможных вариантов решения задачи
ОПК-1 Способен применять	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения

фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности	ОПК-1.2 Способен строить математические модели, производить количественные расчеты и оценки
	ОПК-1.3 Способен определять границы применимости полученных результатов
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.4 Владеет навыками работы с компьютером и компьютерными сетями с целью получения, хранения и обработки научной (технической, технологической) информации
ПК-4 Способен критически оценивать применимость используемых методик и методов	ПК-4.1 Знает численные порядки величин, характерных для соответствующей профессиональной области
	ПК-4.3 Способен обосновать причинно-следственные отношения используемых понятий и моделей

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- основные методы теории систем, свойства систем, основы теории формальных систем и её значение для проблематики алгоритмизации, программирования;
- сущность, важность, актуальность и особенности проблемы обеспечения успешного проектирования программных систем;
- особенности создания программ как объекта обработки и анализа;
- принципы реализации поэтапной разработки программ в ходе их создания;
- критерии качества программных продуктов;
- требования к построению программных систем и рекомендации по обеспечению их функционирования и обслуживания.

уметь:

- эффективно использовать приемы, методы и средства проектирования программных продуктов;
- практически реализовывать полученные навыки разработки приемов, методов программирования;
- формулировать задачи программирования систем, подбирать рациональные способы и средства их реализации.

владеть:

- приемами, методами и средствами создания программных продуктов;
- навыками разработки приемов, методов и средств проектирования программ;
- методиками реализации задач программирования, подбирать рациональные способы и средства их реализации;
- практикой исследования и решения теоретических и практических задач.

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	Введение в программирование систем.	1			1
2	Парадигмы успешного программирования.	1			1

3	Особенности проектирования и обработки протоколов.	1			2
4	Системы контроля версий	2			4
5	Алгоритмы	2			4
6	Рефакторинг и шаблоны проектирования.	2			4
7	Эффективная передача данных по сети связи . Физический уровень Ethernet	2			4
8	Стек TCP/IP	2			4
9	Преимущества агрегированной передачи информации.	1			2
10	Платформенная реализация программных продуктов.	1			4
Итого часов		15			30
Подготовка к экзамену		0 час.			
Общая трудоёмкость		45 час., 1 зач.ед.			

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

Семестр: 7 (Осенний)

1. Введение в программирование систем.

1.1. Модель программной системы.

1.2. Обобщенная структурная схема программной системы, разбиение на функциональные подсистемы.

2. Парадигмы успешного программирования.

2.1. Развитие языков программирования.

2.2. Объектно-ориентированное проектирование программных продуктов.

2.3. Первая парадигма программирования – «море функций».

2.4. Библиотеки программ (подпрограмм, функций). Специализированные библиотеки.

2.5. Модульное программирование – вторая парадигма. Цепочное и каркасное построение программных систем.

2.6. Преимущества и недостатки объектных технологий.

3. Особенности проектирования и обработки протоколов.

3.1. Виды протоколов передачи данных, решаемые ими задачи.

3.2. Применение разных видов при создании новых протоколов.

3.3. Программная обработка протоколов.

4. Системы контроля версий

4.1. Принцип управления проектом, поддержка версий.

4.2. Классический подход к построению программного продукта.

4.3. Системы контроля версий и их роль в разработке.

4.4. Стратегии ветвления.

5. Алгоритмы

5.1. Возможность декомпозиции проекта в отдельные подсистемы.

5.2. Динамическое программирование (описание подхода, применение для решения задач поиска кратчайших путей, поиска наибольшей возрастающей подпоследовательности, и других).

6. Рефакторинг и шаблоны проектирования.

6.1. Сохранение привычной модели организации вычислений.

6.2. Улучшение качества кода.

6.3. Стил кодирования. Учет возможностей специализированной аппаратной обработки кода.

6.4 Распространенные шаблоны проектирования.

7. Эффективная передача данных по сети связи . Физический уровень Ethernet

7.1. Манчестерский код, NRZ, MLT3, кодирование 4B/5B, 8B/10B, PAM-5.

7.2. Применение кодирования для 10Base-T, 100Base-TX, 1000Base-TX, 1000-Base-X (обзор).

7.3. Кодирование 64B/66B. Физический уровень 10GBase-R, 40GBase-LR4, 100GBase-LR4..

8. Стек TCP/IP

8.1. Применение и модификации стандартных протоколов.

8.2. Модель OSI, Ethernet, IP.

8.3. Работа типичной сети.

8.4. Коммутатор, маршрутизатор.

9. Преимущества агрегированной передачи информации.

9.1. Информационные процессы в цифровых системах.

9.2. Методики оптимизации: «тонкая» настройка транспортирующего протокола; мирный spoofing; сжатие заголовков и данных.

9.3. Принципы сжатия: заголовков (VJ); данных (LZ, Huffman, арифметическое кодирование); мода на DRE

9.4. Необходимость согласованного аппаратно-программного сопровождения работы с данными.

10. Платформенная реализация программных продуктов.

10.1. Технология платформенного программирования.

10.2. Комплексование программных продуктов.

10.3. Специализация, сущность и необходимость создания новых платформ.

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

Учебная аудитория, оснащенная мультимедиа проектором, экраном и микрофоном.

Программные и аппаратно-программные макеты систем управления, системы распознавания.

Аппаратные и программные реализации элементарных компонент специализированных систем.

6.Перечень рекомендуемой литературы

Основная литература

1. Информационные системы [Текст], учеб. пособие для вузов /Ю. С. Избачков, В. Н. Петров. -СПб., Питер, 2005

Дополнительная литература

1. Искусство программирования [Текст]. Т. 1, [учеб. пособие для вузов] /Дональд Э. Кнут ; под общ. ред. Ю. В. Козаченко ; [пер. с англ. С. Г. Тригуб и др.] iОсновные алгоритмы. -М., Вильямс, 2002

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

<http://www.intuit.ru> – открытый институт Интуит
<http://lib.mipt.ru> – электронная библиотека МФТИ
<http://www.citforum.ru/hardware/> - библиотека CIT-Forum

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

На лекционных занятиях используются мультимедийные технологии, включая демонстрацию презентаций и действующих макетов.

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

Успешное освоение курса требует напряжённой самостоятельной работы студента. В программе курса приведено минимально необходимое время для работы студента над темой.

Самостоятельная работа включает в себя:

- чтение, анализ и конспектирование рекомендованной литературы,
- проработку учебного материала (по конспектам лекций, учебной и научной литературе), подготовку ответов на вопросы, предназначенных для самостоятельного изучения,
- доказательство отдельных утверждений, свойств;
- решение задач, предлагаемых студентам на лекциях и практических занятиях,
- подготовку к практическим занятиям, коллоквиумам, дифференцированному зачёту.

Руководство и контроль за самостоятельной работой студента осуществляется в форме индивидуальных консультаций. Показателем владения материалом служит умение решать задачи. Для формирования умения применять теоретические знания на практике студенту необходимо решать как можно больше задач. При решении задач каждое действие необходимо аргументировать, ссылаясь на известные теоретические сведения. Значительно облегчить решение задачи может хорошо выполненный чертёж, структурная схема или схема алгоритма если он соответствует условию задачи. При подготовке к занятиям необходимо повторять ранее изученные основные определения, формулировки теорем. В начале занятия, как правило, проводится короткий (10-15 минут) опрос по материалу прошедших занятий в устной или письменной форме. Обычно придерживаются следующей схемы: изучение материала лекции по конспекту в тот же день, когда была прослушана лекция (10-15 минут); повторение материала накануне следующей лекции (10-15 минут), проработка учебного материала по конспектам лекций, учебной и научной литературе, подготовка ответов на вопросы, предназначенных для самостоятельного изучения (1 час неделю), подготовка к практическому занятию, решение задач (1 час). Важно добиться понимания изучаемого материала, а не механического его запоминания. При затруднении изучения отдельных тем, вопросов, следует обращаться за консультациями к лектору или преподавателю, ведущему практические занятия.

Обязательным требованием является выполнение домашних работ.

Промежуточный контроль знаний проводится в виде коллоквиумов, на которых студенту предлагается письменно ответить на теоретический вопрос и решить две задачи по теме коллоквиума, а также студенту в ходе освоения курса необходимо выполнить две домашних индивидуальных работы с их последующей защитой.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению:	Прикладные математика и физика
профиль подготовки:	Радиотехника и компьютерные технологии Физтех-школа Радиотехники и Компьютерных Технологий кафедра электронных вычислительных машин
курс:	4
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 7 (осенний) - Дифференцированный зачет

Разработчики:

Е.Г. Кречетов, канд. техн. наук

В.А. Поздняков, канд. техн. наук

Н.Б. Преображенский, канд. техн. наук, старший научный сотрудник

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
	УК-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки
	УК-1.5 Определяет и оценивает практические последствия возможных вариантов решения задачи
ОПК-1 Способен применять фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения
	ОПК-1.2 Способен строить математические модели, производить количественные расчеты и оценки
	ОПК-1.3 Способен определять границы применимости полученных результатов
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.4 Владеет навыками работы с компьютером и компьютерными сетями с целью получения, хранения и обработки научной (технической, технологической) информации
ПК-4 Способен критически оценивать применимость используемых методик и методов	ПК-4.1 Знает численные порядки величин, характерных для соответствующей профессиональной области
	ПК-4.3 Способен обосновать причинно-следственные отношения используемых понятий и моделей

2. Показатели оценивания компетенций

В результате изучения дисциплины «Программирование систем» обучающийся должен:

знать:

- основные методы теории систем, свойства систем, основы теории формальных систем и её значение для проблематики алгоритмизации, программирования;
- сущность, важность, актуальность и особенности проблемы обеспечения успешного проектирования программных систем;
- особенности создания программ как объекта обработки и анализа;
- принципы реализации поэтапной разработки программ в ходе их создания;
- критерии качества программных продуктов;
- требования к построению программных систем и рекомендации по обеспечению их функционирования и обслуживания.

уметь:

- эффективно использовать приемы, методы и средства проектирования программных продуктов;
- практически реализовывать полученные навыки разработки приемов, методов программирования;
- формулировать задачи программирования систем, подбирать рациональные способы и средства их реализации.

владеть:

- приемами, методами и средствами создания программных продуктов;
- навыками разработки приемов, методов и средств проектирования программ;
- методиками реализации задач программирования, подбирать рациональные способы и средства их реализации;
- практикой исследования и решения теоретических и практических задач.

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

Перечень типовых (примерных) вопросов для подготовки к текущему контролю:

1. Опишите структуру MAC-адреса сети Ethernet.
2. По каким правилам работает коммутатор? Чем опасно наличие колец в топологии сети Ethernet?
3. Опишите назначение обязательных полей в заголовке IPv4. Зачем нужна маска подсети? Как работает протокол MPLS?
4. Опишите назначение полей в заголовке TCP. Как происходит работа с TCP-сессией?
5. Опишите особенности следующих видов фрагментации/дефрагментации данных: по смещению, по порядковому номеру, на фрагменты равной длины.
6. Опишите особенности разграничения передаваемых данных с помощью флагов и бит-стаффинга; опишите машину состояний для поиска синхронизма по кадрам (hunt, presync, sync).
7. Какие основные проблемы решаются с помощью оптимизации трафика? Зачем настраивать размер окна в протоколе TCP? Что такое мирный спуфинг?
8. Когда имеет практический смысл сжимать заголовок пакета? Как используется алгоритм Ван Якобсона для IPv4?
9. Опишите два наиболее распространенных алгоритма сжатия данных без потерь. В чем суть технологии Data Redundancy Elimination (DRE)?
10. В каких областях обычно применяют ППБМ? Сравните между собой ППБМ, ASIC и универсальный процессор (плюсы и минусы каждого решения).
11. Каковы преимущества и недостатки централизованной и распределенной систем управления версиями? Опишите назначение основных ветвей при использовании стратегии ветвления git-flow.
12. Как решить задачу нахождения в последовательности чисел максимальной возрастающей подпоследовательности с помощью динамического программирования?
13. Что такое рефакторинг? В каких случаях он целесообразен?
14. Приведите примеры правильного и неправильного именования переменных и форматирования кода (выравнивание, группировка и т.п.) на C/C++.
15. Предложите методы тестирования приложения, которое ищет в текстовом файле все числа и сохраняет в другой файл те из них, которые больше 100.

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

Перечень контрольных вопросов для дифференцированного зачета:

1. Стиль кодирования.
2. Модель программной системы.
3. Создании новых протоколов.
4. Разбиение проекта на функциональные подсистемы.
5. Алгоритмы. Понятие и классификация.
6. Рефакторинг и шаблоны проектирования.
7. Эффективная передача данных по сети связи.
8. Сохранение привычной модели организации вычислений при программировании больших систем.
9. Улучшение качества кода. Возможности программиста.
10. Учет возможностей специализированной аппаратной обработки кода.
11. Распространенные шаблоны проектирования программ.

12. Парадигмы успешного программирования.
13. Развитие языков программирования.
14. Объектно-ориентированное проектирование программных продуктов.
15. Преимущества и недостатки объектных технологий.
16. Системы контроля версий и их роль в разработке.
17. Особенности проектирования и программной обработки протоколов.
18. Возможность декомпозиции проекта в отдельные подсистемы
19. Виды протоколов передачи данных, решаемые ими задачи.
20. Физический уровень Ethernet.
21. Применение разных видов протоколов
22. Стратегии ветвления при развитии программных проектов.
23. Программная обработка протоколов.
24. Системы контроля версий.
25. Принцип управления проектом, поддержка версий.
26. Классический подход к построению программного продукта.
27. Динамическое программирование (описание подхода)
28. Решение динамических задач поиска кратчайших путей, поиска наибольшей возрастающей подпоследовательности, и других.
29. Манчестерский код, NRZ, MLT3, кодирование 4B/5B, 8B/10B, PAM-5.
30. Применение кодирования для 10Base-T, 100Base-TX, 1000Base-TX, 1000-Base-X (обзор).
31. Кодирование 64B/66B. Физический уровень 10GBase-R, 40GBase-LR4, 100GBase-LR4..
32. Стек TCP/IP
33. Применение и модификации стандартных протоколов.
34. Модель OSI, Ethernet, IP.
35. Работа типичной сети.
36. Коммутатор, маршрутизатор – программирование и управление.
37. Методики оптимизации: «тонкая» настройка транспортирующего протокола; мирный spoofing; сжатие заголовков и данных.
38. Принципы сжатия: заголовков (VJ); данных (LZ, Huffman, арифметическое кодирование); мода на DRE
39. Необходимость согласованного аппаратно-программного сопровождения работы с данными.
40. Платформенная реализация программных продуктов.
41. Преимущества агрегированной передачи информации
42. Информационные процессы в цифровых системах.
43. Программные и аппаратные возможности компьютера при обработке данных
44. Особенности построения систем передачи цифровых данных.
45. Недостатки программной обработки потока данных.
46. Реализация передачи данных с помощью средств программирования низкого уровня.
47. Использование моделей подсистем при создании больших программных систем.

Критерии оценивания

Оценка отлично (10) 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) 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) выставляется студенту, показавшему 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) 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) 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) 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) 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) 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) 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) 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

Дифференцированный зачет проводится в устной форме.

При проведении дифференцированного зачета обучающемуся предоставляется 20 минут на подготовку. Опрос обучающегося проводится в течение 30 минут.

Во время проведения экзамена и зачета обучающиеся могут пользоваться программой дисциплины.