

Утверждено на заседании  
Академического совета по программной инженерии  
факультета компьютерных наук  
(протокол от 13 декабря 2024 г., № 2.3-09.2/131224-4)

**Федеральное государственное автономное образовательное учреждение  
высшего образования**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Программа собеседования  
для поступающих на образовательную  
программу магистратуры  
*«Системное программирование»*

09.04.04. Программная инженерия

Академический руководитель программы  Петренко А.К.

Москва, приемная кампания 2025 года

# Введение

В основу настоящей программы положены следующие дисциплины: математические основы программирования; вычислительные машины, системы и сети; языки и системы программирования; технология разработки программного обеспечения; операционные системы; методы хранения и доступа к данным, организация баз данных и знаний; защита данных и программных систем.

Программа разработана кафедрой системного программирования факультета Компьютерных наук НИУ ВШЭ.

## 1. Математические основы программирования

1. Понятие алгоритма и его уточнения: машины Тьюринга, нормальные алгоритмы Маркова, рекурсивные функции.
2. Алгебра логики. Булевы функции, канонические формы задания булевых функций. Понятие полной системы. Критерий полноты Поста. Минимизация булевых функций в классах нормальных форм.
3. Исчисление предикатов первого порядка. Понятие интерпретации. Выполнимость и общезначимость формулы первого порядка. Понятие модели. Теорема о полноте исчисления предикатов первого порядка.
4. Отношения и функции. Отношение эквивалентности и разбиения. Фактор множества. Отношения частичного порядка. Теоретико-множественное и алгебраическое определения решетки, их эквивалентность. Свойства решеток. Булевы решетки. Полные решетки.
5. Формальные языки и способы их описания. Классификация формальных грамматик. Их использование в лексическом и синтаксическом анализе.
6. Основы комбинаторного анализа. Метод производящих функций, метод включений и исключений. Примеры применения.
7. Коды с исправлением ошибок. Алфавитное кодирование. Методы сжатия информации.

## 2. Вычислительные машины, системы и сети

1. Архитектура современных компьютеров. Организации памяти и архитектура процессора современных вычислительных машин. Страничная и сегментная организация виртуальной памяти. Кэш-память. Командный и арифметический конвейеры, параллельное выполнение независимых команд, векторные команды. Специализированные процессоры. Машины, обеспечивающие выполнение вычислений, управляемых потоком данных. Организация ввода-вывода, устройства сопряжения с объектами.
2. Классификация вычислительных систем (ВС) по способу организации параллельной обработки. Многопроцессорные и многомашинные комплексы. Вычислительные кластеры. Проблемно-ориентированные параллельные структуры: матричные ВС, систолические структуры, нейросети.
3. Назначение, архитектура и принципы построения информационно-вычислительных сетей (ИВС). Локальные и глобальные ИВС, технические и программные средства объединения различных сетей.
4. Методы и средства передачи данных в ИВС, протоколы передачи данных.
5. Особенности архитектуры локальных сетей (Ethernet, Token Ring, FDDI).
6. Сеть Internet, доменная организация, семейство протоколов TCP/IP. Информационно-вычислительные сети и распределенная обработка информации.

## 3. Языки и системы программирования. Технология разработки программного обеспечения

1. Языки программирования. Процедурные языки программирования (С), Функциональные языки программирования (Lisp, Haskell), объектно-ориентированные языки программирования (Java, C++).
2. Процедурные языки программирования. Основные управляющие конструкции, структура программы. Работа с данными: переменные и константы, типы данных (булевский, целочисленные, плавающие, символьные, типы диапазона и перечисления, указатели), структуры данных (массивы и записи). Процедуры (функции): вызов процедур, передача параметров (по ссылке, по значению, по результату), локализация переменных, побочные эффекты. Обработка исключительных ситуаций. Библиотеки процедур и их использование.

3. Объектно-ориентированное программирование. Классы и объекты, наследование, интерфейсы. Понятие об объектном окружении. Рефлексия. Библиотеки классов. Средства обработки объектов (контейнеры и итераторы).
4. Основы построения трансляторов. Структура оптимизирующего транслятора. Промежуточные представления программы: последовательность символов, последовательность лексем, синтаксическое дерево, абстрактное синтаксическое дерево. Уровни промежуточного представления: высокий, средний, низкий. Формы промежуточного представления.
5. Анализ исходной программы в компиляторе. Автоматные (регулярные) грамматики и сканирование, контекстно свободные грамматики и синтаксический анализ, организация таблицы символов программы, имеющей блочную структуру, хеш-функции. Нисходящие (LL(1)-грамматики) и восходящие (LR(1)-грамматики) методы синтаксического анализа. Атрибутные грамматики и семантические программы, построение абстрактного синтаксического дерева. Автоматическое построение лексических и синтаксических анализаторов по формальным описаниям грамматик. Системы lex и yacc. Система Gentle.
6. Оптимизация программ. Основные методы оптимизации. Машиннозависимая и машиннонезависимая оптимизации.
7. Машинно-ориентированные языки, язык ассемблера. Представление машинных команд и констант. Команды транслятору. Их типы, принципы реализации. Макросредства, макровыводы, языки макроопределений, условная макрогенерация, принципы реализации.
8. Системы программирования (СП), типовые компоненты СП: языки, трансляторы, редакторы связей, отладчики, текстовые редакторы. Модульное программирование. Типы модулей. Связывание модулей по управлению и данным.
9. Технология разработки и сопровождения программ. Жизненный цикл программы. Этапы разработки, степень и пути их автоматизации.
10. Отладка, тестирование, верификация и оценивание сложности программ. Генерация тестов. Метрики полноты тестового покрытия.

## 4. Операционные системы

1. Режимы функционирования вычислительных систем, структура и функции операционных систем. Основные блоки и модули. Основные средства аппаратной поддержки функций операционных систем (ОС): система прерываний, защита памяти, механизмы преобразования адресов в системах виртуальной памяти, управление каналами и периферийными устройствами.
2. Виды процессов и управления ими в современных ОС. Представление процессов, их контексты, иерархии порождения, состояния и взаимодействие. Многозадачный (многопрограммный) режим работы. Команды управления процессами. Средства взаимодействия процессов. Модель клиент-сервер и ее реализация в современных ОС.
3. Параллельные процессы, схемы порождения и управления. Организация взаимодействия между параллельными и асинхронными процессами: обмен сообщениями, организация почтовых ящиков. Критические участки, примитивы взаимоисключения процессов, семафоры Дейкстры и их расширения. Проблема тупиков при асинхронном выполнении процессов, алгоритмы обнаружения и предотвращения тупиков.
4. Операционные средства управления процессами при их реализации на параллельных и распределенных вычислительных системах и сетях: стандарты и программные средства MPI, OpenMP, POSIX .
5. Одноуровневые и многоуровневые дисциплины циклического обслуживания процессов на центральном процессоре, выбор кванта.
6. Управление доступом к данным. Файловая система, организация, распределение дисковой памяти. Управление обменом данными между дисковой и оперативной памятью. Рабочее множество страниц (сегментов) программы, алгоритмы его определения.
7. Управление внешними устройствами.
8. Оптимизация многозадачной работы компьютеров. Операционные системы Windows, Unix, Linux. Особенности организации, предоставляемые услуги пользовательского взаимодействия.
9. Операционные средства управления сетями. Эталонная модель взаимодействия открытых систем ISO/OSI. Маршрутизация и управление потоками данных в сети. Локальные и глобальные сети.

## 5. Методы хранения данных и доступа к ним. Организация баз данных и знаний

10. Концепция типа данных. Абстрактные типы данных. Объекты (основные свойства и отличительные признаки).
11. Основные структуры данных, алгоритмы обработки и поиска. Сравнительная характеристика методов хранения и поиска данных.
12. Основные понятия реляционной и объектной моделей данных.
13. Теоретические основы реляционной модели данных. Реляционная алгебра, реляционное исчисление. Функциональные зависимости и нормализация отношений.
14. CASE-средства и их использование при проектировании базы данных (БД).
15. Организация и проектирование физического уровня БД. Методы индексирования.
16. Обобщенная архитектура, состав и функции системы управления базой данных (СУБД). Характеристика современных технологий БД. Примеры соответствующих СУБД.
17. Основные принципы управления транзакциями, журнализацией и восстановлением.
18. Язык баз данных SQL. Средства определения и изменения схемы БД, определения ограничений целостности. Контроль доступа. Средства манипулирования данными.
19. Стандарты языков SQL. Интерактивный, встроенный, динамический SQL.
20. Основные понятия технологии клиент-сервер. Характеристика SQL-сервера и клиента. Сетевое взаимодействие клиента и сервера.

### Основная литература

1. Ахо А.В., Лам М.С., Сети Р., Ульман Дж.Д. Компиляторы. Принципы, технологии и инструментарий. - М.: Вильямс, 2016.
2. Воеводин В.В., Воеводин Вл. В. Параллельное программирование. - СПб.: БХВ- Петербург, 2002.
3. Галатенко В.А. Основы информационной безопасности. Под ред. В.Б. Бетелина. - М.: ИНТУИТ.РУ, 2003.
4. Галатенко В.А. Стандарты информационной безопасности. Под ред. В.Б. Бетелина. - М.: ИНТУИТ.РУ, 2004.
5. Дейт К.Дж. Введение в системы баз данных. - М.: Вильямс, 2018.
6. Кнут Д. Искусство программирования. Т. 1 – 3. - М. Вильямс, 2017..
7. Кузнецов С.Д. Базы данных: языки и модели. Учебник. М.: Бином-Пресс, 2008.
8. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы, построение и анализ. - М.: Вильямс, 2020.
9. Липаев В.В. Программная инженерия. Методологические основы. М.: Государственный Университет – Высшая школа экономики, 2006.
10. Стивенс Р., Раго С. UNIX. Профессиональное программирование. - СПб.: Символ-Плюс, 2007.
11. Судоплатов С.В., Овчинникова Е.В. Дискретная математика. Инфра-М, Новосибирский государственный технический университет (НГТУ), 2009.
12. Танненбаум Э., Уэзеролл Д. Компьютерные сети. СПб.: Питер, 2016.
13. Таненбаум Э. Современные операционные системы. - СПб.:Питер, 2002.
14. Таненбаум Э., Ван Стен М. Распределенные системы. Принципы и парадигмы. - СПб.:Питер, 2003.
15. Танненбаум Э. Архитектура компьютера. СПб.: Питер, 2006.
16. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. - М.: Вильямс, 2016.
17. Яблонский С.В. Введение в дискретную математику. - М.: Наука, 2001.

### Дополнительная литература

1. Гласс Г., Эйбле К. Unix для программистов и пользователей. СПб.: БХВ-Петербург, 2004.
2. Корнеев В.В. Вычислительные системы. М.: Гелиос АРВ, 2004.
3. Королев Л.Н. Архитектура процессоров электронных вычислительных машин, М.: Издательский отдел ВМиК МГУ, 2003.
4. Хаулет Т. Защитные средства с открытыми исходными текстами. Пер. с англ. - М.: ИНТУИТ.РУ, БИНОМ. Лаборатория знаний, 2007.

5. Арлоу Д., Нейштадт А. UML 2 и унифицированный процесс. Практический объектно-ориентированный анализ и проектирование. - СПб.: Символ-Плюс. - 2008.
6. Рамбо Дж., Блаха М. UML 2.0. Объектно-ориентированное моделирование и разработка. - СПб.: Питер. - 2007.
7. Гамма Э. и др. Приемы объектно-ориентированного проектирования. Паттерны проектирования.: Пер. с англ. - СПб.: Питер, 2016.
8. Б.Мейер. Объектно-ориентированное конструирование программных систем – М.: Русская Редакция, 2005.
9. Д. Месарош. Шаблоны тестирования xUnit. М.: Вильямс, 2008
10. Материалы курса В.В.Кулямина "Тестирование программного обеспечения": <http://mbt-course.narod.ru> (доступ 19.01.2017).
11. Ю.Г. Карпов. Model Checking. Верификация параллельных и распределенных программных систем. — СПб.: БХВ-Петербург, 2010.

## Формат собеседования

Собеседование проводится на русском языке.

Собеседование состоит из двух частей. В первой части абитуриент рассказывает о себе, о мотивах, которыми он руководствуется, выбирая системное программирование как направление своего обучения и дальнейшей профессиональной деятельности. На это планируется 5-10 минут.

Затем студент получает 2-3 вопроса и, если нужно, берет время на подготовку. На подготовку и ответ планируется 30-40 минут.