**Санкт-Петербургский государственный университет**

**Р А Б О Ч А Я П Р О Г Р А М М А**

**УЧЕБНОЙ ДИСЦИПЛИНЫ**

Организация и дизайн современных компьютеров

Modern Computer Structure and Design

**Язык(и) обучения**

русский

Трудоемкость в зачетных единицах: 3

Регистрационный номер рабочей программы: 025200

**Раздел 1. Характеристики учебных занятий**

**1.1. Цели и задачи учебных занятий**

Целью дисциплины является обучение основным концепциям построения современной вычислительной техники; методами количественной оценки свойств компьютерных архитектур; основам разработки и тестирования новых компьютерных архитектур; принципам взаимодействия компьютерной архитектуры с программным обеспечением.   
Основным методологическим принципом построения программы курса, равно как и всей   
концепции обучения в целом, является принцип поэтапного системного накопления   
знаний и формирования необходимых компетенций по модели: от простого и/или   
знакомого к сложному и/или незнакомому. Основной методологической стратегией   
прохождения отдельных разделов программы является ступенчатость и цикличность,   
предусматривающая постепенный возврат к ранее усвоенному материалу на более   
высоком концептуальном уровне, с учетом исторической перспективы и количественного и качественного сравнения.

По окончании обучения обучающиеся должны знать содержимое курса «Организация и дизайн современных компьютеров», владеть терминологией, достаточно полно представлять возможности существующих и предшествующих компьютерных архитектур, уметь выбрать архитектуру в соответствии задачей и оценивать ее количественные и качественные свойства в рамках задачи.

**1.2. Требования подготовленности обучающегося к освоению содержания учебных занятий (пререквизиты)**

Программа дисциплины рассчитана на обучающихся 3–го курса. Максимальная   
эффективность программы будет обеспечена при следующем условии: обучающийся владеет базовыми математическими понятиями и базовыми понятиями в области информатики и теории алгоритмов, изученными на первом и втором курсах. Также рекомендуется наличие базовых знаний в области теории построения современных операционных систем.

**1.3. Перечень результатов обучения (learning outcomes)**

* владеть терминологией;
* достаточно полно представлять возможности существующих и предшествующих компьютерных архитектур включая (CPU, SMP, NUMA, GPU, IoT, SoC);
* уметь выбрать архитектуру в соответствии задачей;
* оценивать количественные и качественные свойства архитектуры в рамках задачи.

Дисциплина способствует развитию следующих компетенций:

* ОПК-1 — способен применять фундаментальные знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности;
* ОПК-3 — способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения;
* ОПК-4 — способен участвовать в разработке технической документации программных продуктов и программных комплексов;
* ОПК-5 — способен инсталлировать и сопровождать программное обеспечение для информационных систем и баз данных, в том числе отечественного производства;
* ПКА-1 — способен демонстрировать базовые знания математических и естественных наук, программирования и информационных технологий;
* ПКП-1 — способность проводить под научным руководством исследование на основе существующих методов в конкретной области профессиональной деятельности;
* ПКП-2 — способен решать задачи в области развития науки, техники и технологии с учетом нормативного правового регулирования в сфере интеллектуальной собственности;
* ПКП-4 — способен применять современные информационные технологии при проектировании, реализации, оценке качества и анализа эффективности программного обеспечения для решения задач в различных предметных областях;
* ПКП-5 — способен использовать основные методы и средства автоматизации проектирования, реализации, испытаний и оценки качества при создании конкурентоспособного программного продукта и программных комплексов, а также способен использовать методы и средства автоматизации, связанные с сопровождением, администрированием и модернизацией программных продуктов и программных комплексов;
* ПКП-6 — способен использовать знания направлений развития компьютеров с традиционной (нетрадиционной) архитектурой; современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ; тенденции развития функций и архитектур проблемно-ориентированных программных систем и комплексов в профессиональной деятельности;
* ПКП-8 — способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования;
* УКБ-3 — способен понимать сущность и значение информации в развитии общества, использовать основные методы получения и работы с информацией с учетом современных технологий цифровой экономики и информационной безопасности.

**1.4. Перечень и объём активных и интерактивных форм учебных занятий**

Активные и интерактивные формы учебных занятий общим объёмом 4 ак. часа состоят из лекций, предполагающих активное обсуждение материала с преподавателем.

**Раздел 2. Организация, структура и содержание учебных занятий**

**2.1. Организация учебных занятий**

**2.1.1 Основной курс**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Трудоёмкость, объёмы учебной работы и наполняемость групп обучающихся | | | | | | | | | | | | | | | | | | |
| Код модуля в составе дисциплины,  практики и т.п. | Контактная работа обучающихся с преподавателем | | | | | | | | | | | | Самостоятельная работа | | | | Объём активных и интерактивных  форм учебных занятий | Трудоёмкость |
| лекции | семинары | консультации | практические  занятия | лабораторные работы | контрольные работы | коллоквиумы | текущий контроль | промежуточная  аттестация | итоговая аттестация | под руководством преподавателя | в присутствии  преподавателя | сам. раб. с использованием  методических материалов | текущий контроль (сам.раб.) | промежуточная аттестация (сам.раб.) | итоговая аттестация  (сам.раб.) |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | | | | | | | | | | | | | |
| Форма обучения: очная | | | | | | | | | | | | | | | | | | |
| Семестр 6 | 30 |  | 2 |  |  |  |  |  | 2 |  |  |  | 46 |  | 28 |  | 4 | 3 |
|  | 2-30 |  | 2-30 |  |  |  |  |  | 1-30 |  |  |  | 1-1 |  | 1-1 |  |  |  |
| ИТОГО | 30 |  | 2 |  |  |  |  |  | 2 |  |  |  | 46 |  | 28 |  |  | 3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Виды, формы и сроки текущего контроля успеваемости и промежуточной аттестации | | | | | | |
| Код модуля в составе дисциплины, практики и т.п. | Формы текущего контроля успеваемости | | Виды промежуточной аттестации | | Виды итоговой аттестации  (только для программ итоговой аттестации и дополнительных образовательных программ) | |
| Формы | Сроки | Виды | Сроки | Виды | Сроки |
| ОСНОВНАЯ ТРАЕКТОРИЯ | | | | | | |
| Форма обучения: очная | | | | | | |
| Семестр 6 |  |  | экзамен, устно, традиционная форма | по графику промежуточной аттестации |  |  |

**2.2. Структура и содержание учебных занятий**

1. **Количественный подход к компьютерным архитектурам**

Иерархическая организация вычислительных устройств. Классы компьютеров. Определение компьютерной архитектуры. Измерение производительности и других характеристик компьютера. Основные формулы и модели оценки. Основные параметры: производительность, энергоэффективность, стоимость.

1. **История развития компьютерных архитектур от машины Бебиджа до сегодняшних дней и основные принципы управления сложностью.**

Поколение 0: Механические компьютеры. 1: Вакуумные лампы. 2: Транзисторы, 3: Интегральные схемы. 4: БИС. 5: малопотребляющие и проникающие вычисления. Современные семейства микропроцессоров: x86, ARM, AVR.

1. **Архитектура компьютера. Процессор**

Внутренняя структура процессора. Инструкции. RISC и CISC. Основные принципы построения. Параллелизм: Superscalar, VLIW, GPU, SIMD, Multi-Core, VIP.

1. **Архитектура компьютера. Оперативная память**

Память с произвольным доступом. Энергозависимая и энергонезависимая память. Типы памяти: ROM, EPROM, SRAM, DRAM, NOR и NAND Flash. Порядок байтов. Коды коррекции ошибок. Кэширование. Типы упаковки и интерфейсы.

1. **Архитектура компьютера. Внешняя память и Ввод/Вывод**

Прерывания. DMA. Внутренние и внешние шины и интерфейсы. Межпроцессорные интерфейсы: HyperTransport, DMI, QuickPath. Внутренние интерфейсы: PCIE, PCI, ISA. Накопители с произвольным доступом, дисковые интерфейсы: SAS, SATA, ATA. Твердотельные накопители. Магнитные накопители. Оптические диски. Внешние интерфейсы: USB, FireWire, ESATA, HDMI, DisplayPort, LPT, COM. Периферийные устройства.

1. **Цифровые логические схемы**

Вентили. Базовые компоненты. Устройство и организация памяти. Процессоры и шины. Примеры: Intel Core2, TI OMAP4430, ATmega168, PCI, USB.

1. **Микроархитектура**

Управляющий автомат (Control path). Операционный автомат (Datapath). Предвыборка. Конвейер. Кэширование. Предсказание переходов. Спекулятивное исполнение.

1. **Примеры микроархитектуры микропроцессоров**

Intel Core2, TI OMAP4430, ATmega168.

1. **Архитектура инструкций**

Модели памяти. Регистры. Типы инструкций. Адресация. Формат инструкций. Примеры: Intel Core2, TI OMAP4430, ATmega168.

1. **Операционные системы и системное программирование**

Виртуальная память. Защита памяти. Многозадачность. Обработка прерываний. Ввод/Вывод. Параллелизм и синхронизация.

1. **Средства разработки**

ABI. Компиляторы языков высокого уровня. Ассемблер. Объектные файлы. Отладочная информация. Линкер. Загрузка и динамические библиотеки. Кросс-компиляция.

1. **Параллелизм и параллельные архитектуры**

Параллелизм внутри процессора. Акселераторы и сопроцессоры. Параллельные архитектуры с общей памятью. Параллельные архитектуры на основе обмена сообщениями. Распределенные вычисления.

1. **Архитектура компьютера с точки зрения прикладного программирования**

Целевые оптимизации в языках высокого уровня. Кэширование и локальность доступа к данным. Профилирование.

1. **Архитектура и программирование GP GPU**

Микроархитектура. Конвейер отрисовки OpenGL 3.0 и DX12. Примеры архитектур: NVIDIA Pascal, AMD RX400, Intel HD Graphics. Интерфейсы программирования: Cuda, OpenCL, OpenGL SL, DX HLSL.

1. **Непроцессорные архитектуры, системы на кристалле (SoC), встроенные системы**

Программируемые логические схемы (FPGA), специализированные логические схемы (ASIC), синтез логических схем из языков высокого уровня (HLS), малопотребляющие системы, интернет вещей (IoT), специфика программирования встроенных систем.

**Раздел 3. Обеспечение учебных занятий**

**3.1. Методическое обеспечение**

**3.1.1 Методические указания по освоению дисциплины**

Методические материалы включают в себя следующие типы материалов: учебные пособия, электронные учебные пособия, Интернет-ресурсы.

**3.1.2 Методическое обеспечение самостоятельной работы**

Основная и дополнительная литература.

**3.1.3 Методика проведения текущего контроля успеваемости и промежуточной аттестации и критерии оценивания**

Контроль успеваемости обучающихся осуществляется посредством проводимого в конце семестра устного экзамена.

*Методика проведения экзамена.*

Экзамен проводится в устной форме. Билет состоит из двух вопросов, на подготовку ответа на которые даётся не менее одного академического часа (при подготовке можно пользоваться литературой). После ответа на вопросы билета преподаватель вправе задать дополнительные вопросы по любой теме из списка вопросов, вынесенных на экзамен. Количество и содержание дополнительных вопросов – на усмотрение преподавателя, принимающего экзамен. Каждый ответ оценивается по шкале от 0 (нет ответа) до 5 (очень хороший ответ), результирующая оценка получается следующим образом:

1. Оценки за ответы на два основных вопроса усредняются
2. Оценки за ответы на дополнительные вопросы усредняются, результат складывается с баллами, полученными в п.1.
3. Результат умножается на 10, чтобы получить итоговый процент усвоения дисциплины (в диапазоне от 0% до 100%).

Далее применяется стандартная шкала оценивания ECTS:

|  |  |  |
| --- | --- | --- |
| Процент освоения дисциплины | Оценка ECTS | Оценка при проведении экзамена |
| от 90% до 100% | A | отлично |
| от 80% до 89% | B | хорошо |
| от 70% до 79% | C | хорошо |
| от 60% до 69% | D | удовлетворительно |
| от 50% до 59% | E | удовлетворительно |
| менее 50% | F | неудовлетворительно |

**3.1.4 Методические материалы для проведения текущего контроля успеваемости и промежуточной аттестации (контрольно-измерительные материалы, оценочные средства)**

*Список примерных вопросов для устного экзамена:*

1. Иерархическая организация вычислительных устройств. Классы компьютеров. Определение компьютерной архитектуры.
2. Измерение производительности и других характеристик компьютера. Основные формулы и модели оценки. Основные параметры: производительность, энергоэффективность, стоимость.
3. История развития компьютерных архитектур: поколение 0: Механические компьютеры. 1: Вакуумные лампы. 2: Транзисторы,
4. История развития компьютерных архитектур: поколение 3: Интегральные схемы. 4: БИС. 5: малопотребляющие и проникающие вычисления. Современные семейства микропроцессоров: x86, ARM, AVR.
5. Внутренняя структура процессора. Инструкции.
6. RISC и CISC. Основные принципы построения.
7. Параллелизм: Superscalar, VLIW, GPU, SIMD, Multi-Core, VIP.
8. Память с произвольным доступом. Энергозависимая и энергонезависимая память. Типы памяти: ROM, EPROM, SRAM, DRAM, NOR и NAND Flash.
9. Порядок байтов. Коды коррекции ошибок. Кэширование. Типы упаковки и интерфейсы.
10. Прерывания. DMA. Внутренние и внешние шины и интерфейсы.
11. Межпроцессорные интерфейсы: HyperTransport, DMI, QuickPath.
12. Внутренние интерфейсы: PCIE, PCI, ISA. Накопители с произвольным доступом, дисковые интерфейсы: SAS, SATA, ATA.
13. Твердотельные накопители. Магнитные накопители. Оптические диски.
14. Внешние интерфейсы: USB, FireWire, ESATA, HDMI, DisplayPort, LPT, COM. Периферийные устройства.
15. Вентили. Базовые компоненты. Устройство и организация памяти.
16. Процессоры и шины. Примеры: Intel Core2, TI OMAP4430.
17. Процессоры и шины. Примеры: ATmega168, PCI, USB.
18. Управляющий автомат (Control path). Операционный автомат (Datapath).
19. Предвыборка. Конвейер. Кэширование. Предсказание переходов. Спекулятивное исполнение.
20. Примеры микроархитектуры микропроцессоров: Intel Core2
21. Примеры микроархитектуры микропроцессоров: TI OMAP4430
22. Примеры микроархитектуры микропроцессоров: ATmega168.
23. Модели памяти. Регистры. Типы инструкций. Адресация. Формат инструкций.
24. Примеры инструкций: Intel Core2, TI OMAP4430, ATmega168.
25. Виртуальная память. Защита памяти.
26. Многозадачность. Обработка прерываний.
27. Ввод/Вывод. Параллелизм и синхронизация.
28. ABI. Компиляторы языков высокого уровня.
29. Ассемблер. Объектные файлы. Отладочная информация.
30. Линкер. Загрузка и динамические библиотеки. Кросс-компиляция.
31. Параллелизм внутри процессора. Акселераторы и сопроцессоры.
32. Параллельные архитектуры с общей памятью.
33. Параллельные архитектуры на основе обмена сообщениями. Распределенные вычисления.
34. Целевые оптимизации в языках высокого уровня. Кэширование и локальность доступа к данным.
35. Профилирование.
36. Микроархитектура GPGPU. Конвейер отрисовки OpenGL 3.0 и DX12.
37. Примеры архитектур GPGPU: NVIDIA Pascal, AMD RX400, Intel HD Graphics.
38. Интерфейсы программирования: Cuda, OpenCL, OpenGL SL, DX HLSL.
39. Программируемые логические схемы (FPGA), специализированные логические схемы (ASIC).
40. Синтез логических схем из языков высокого уровня (HLS), малопотребляющие системы, интернет вещей (IoT), специфика программирования встроенных систем.

**3.1.5 Методические материалы для оценки обучающимися содержания и качества учебного процесса**

Оценка обучающимися содержания и качества учебного процесса по дисциплине осуществляется в установленном в СПбГУ порядке.

**3.2. Кадровое обеспечение**

**3.2.1 Образование и (или) квалификация штатных преподавателей и иных лиц, допущенных к проведению учебных занятий**

К преподаванию дисциплины могут быть допущены преподаватели, имеющие диплом о высшем образовании по соответствующему направлению.  
  
**3.2.2 Обеспечение учебно-вспомогательным и (или) иным персоналом**

Специальных требований нет.

**3.3. Материально-техническое обеспечение**

**3.3.1 Характеристики аудиторий (помещений, мест) для проведения занятий**

В аудиториях, где проводятся занятия, необходимо наличие мультимедийного проектора, экрана, а также доски.

**3.3.2 Характеристики аудиторного оборудования, в том числе неспециализированного компьютерного оборудования и программного обеспечения общего пользования**

Специальных требований нет.

**3.3.3 Характеристики специализированного оборудования**

Специальных требований нет.

**3.3.4 Характеристики специализированного программного обеспечения**

Специальных требований нет.

**3.3.5 Перечень и объёмы требуемых расходных материалов**

Расходные материалы не требуются.

**3.4. Информационное обеспечение**

**3.4.1 Список обязательной литературы**

1. Харрис Д.М. и Харрис С.Л. Цифровая схемотехника и архитектура компьютера:[пер. с англ.]. – бесплатное электронное издание, 2013   
http://easyelectronics.ru/files/Book/digital-design-and-computer-architecture-russian-translation.pdf

**3.4.2 Список дополнительной литературы**

1. Таненбаум Э. С. Архитектура компьютера: [пер. с англ.]. – СПб: Издательский дом" Питер", 2011. Hennessy J. L., Patterson D. A. Computer architecture: a quantitative approach. – Elsevier, 2011.

2. Patterson D. A., Hennessy J. L. Computer organization and design: the hardware/software interface. – Newnes, 2013.

3. Flynn M. J., Luk W. Computer system design: system-on-chip. – John Wiley & Sons, 2011.

4. Betz V., Rose J., Marquardt A. Architecture and CAD for deep-submicron FPGAs. – Springer Science & Business Media, 2012. – Т. 497.

5. Gajski D. D. et al. High—Level Synthesis: Introduction to Chip and System Design. – Springer Science & Business Media, 2012.

**3.4.3 Перечень иных информационных источников**

Не требуется.

**Раздел 4. Разработчики программы**

Салищев Сергей Игоревич, к. ф.-м. н, старший преподаватель кафедры информатики, s.salischev@spbu.ru