**Санкт-Петербургский государственный университет**

**Р А Б О Ч А Я П Р О Г Р А М М А**

**УЧЕБНОЙ ДИСЦИПЛИНЫ**

Архитектура процессора

Processor Architecture

**Язык(и) обучения**

русский

Трудоемкость в зачетных единицах: 4

Регистрационный номер рабочей программы: 002905

**Раздел 1. Характеристики учебных занятий**

**1.1. Цели и задачи учебных занятий**

Цель курса – дать обучающимся представление о внутреннем устройстве процессора, архитектуре компьютера, программировании на языке ассемблера, наборах инструкций современных процессоров.

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

Программа курса предназначена для студентов третьего курса и рассчитана на студентов, изучавших программирование (владеющих терминологией и имеющих практические навыки, включая основы разработки интерфейсов прикладных программ), алгоритмы и структуры данных, архитектуру ЭВМ в объеме 2 курсов.

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

• Представление об архитектуре современных компьютеров.  
• Представление о внутреннем устройстве современных процессоров.  
• Навыки программирования на языке ассемблера под архитектуру Intel 64.

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

Аудиторная учебная работа: теоретические занятия в объеме 2 часов в неделю.

**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 5 | 32 |  | 2 |  |  |  |  |  | 2 |  |  |  | 72 |  | 36 |  | 4 | 4 |
|  | 2-100 |  | 2-100 |  |  |  |  |  | 2-100 |  |  |  | 1-1 |  | 1-1 |  |  |  |
| ИТОГО | 32 |  | 2 |  |  |  |  |  | 2 |  |  |  | 72 |  | 36 |  |  | 4 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 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 5 |  |  | экзамен, устно, традиционная форма | по графику промежуточной аттестации |  |  |

**2.2. Структура и содержание учебных занятий**

1. Введение. Архитектура компьютера, основные понятия.
2. Арифметика целых чисел.
3. Манипуляции с битами и конструирование сложных операций.
4. Операции с многочленами и вычисления в конечных полях .
5. Арифметика чисел с плавающей точкой.
6. Базовая среда исполнения Intel 64.
7. Понятие об ассемблере.
8. Команды пересылки данных.
9. Команды передачи управления.
10. Арифметические и логические команды общего назначения.
11. Арифметическое устройство для вычислений с плавающей точкой (x87 FPU).
12. Технология Intel® MMX™.
13. Технология SSE. Форматы данных и среда исполнения.
14. Технология SSE. Арифметика целых чисел.
15. Технология SSE. Арифметика чисел с плавающей точкой.
16. Расширения AVX и FMA.
17. Вычисления в конечных полях в архитектуре Intel 64.
18. Поддержка шифрования.
19. Сегментное преобразование адреса.
20. Страничное преобразование адреса.
21. Передача управления между кодовыми сегментами. Поддержка многозадачности.
22. Прерывания и исключения.
23. Управление кэшированием.
24. Организация ввода-вывода.
25. Поддержка многопроцессорных систем.
26. Поддержка виртуальных машин. Общий обзор.

**Раздел 3. Обеспечение учебных занятий**

**3.1. Методическое обеспечение**

**3.1.1 Методические указания по освоению дисциплины**

Успешное освоение дисциплины возможно благодаря посещению занятий и самостоятельной работе, включающей в себя чтение специальной литературы по разделам программы.

**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**

*Пример списка вопросов для устного экзамена:*

1. Введение. Архитектура компьютера, основные понятия.
2. Арифметика целых чисел.
3. Манипуляции с битами и конструирование сложных операций.
4. Операции с многочленами и вычисления в конечных полях .
5. Арифметика чисел с плавающей точкой.
6. Базовая среда исполнения Intel 64.
7. Понятие об ассемблере.
8. Команды пересылки данных.
9. Команды передачи управления.
10. Арифметические и логические команды общего назначения.
11. Арифметическое устройство для вычислений с плавающей точкой (x87 FPU).
12. Технология Intel® MMX™.
13. Технология SSE. Форматы данных и среда исполнения.
14. Технология SSE. Арифметика целых чисел.
15. Технология SSE. Арифметика чисел с плавающей точкой.
16. Расширения AVX и FMA.
17. Вычисления в конечных полях в архитектуре Intel 64.
18. Поддержка шифрования.
19. Сегментное преобразование адреса.
20. Страничное преобразование адреса.
21. Передача управления между кодовыми сегментами. Поддержка многозадачности.
22. Прерывания и исключения.
23. Управление кэшированием.
24. Организация ввода-вывода.
25. Поддержка многопроцессорных систем.
26. Поддержка виртуальных машин. Общий обзор.

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

Контроль успеваемости студентов 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:

Оценки за ответы на два основных вопроса усредняются, результат усреднения делится на два.

Оценки за ответы на дополнительные вопросы усредняются, результат усреднения делится на два и складывается с баллами, полученными в п.1.

Если результирующая оценка

* 1. в диапазоне от 2.5 до 3.5, за экзамен ставится «удовлетворительно»;
  2. в диапазоне от 3.5 до 4.5, за экзамен ставится «хорошо»;
  3. в диапазоне от 4.5 до 5, за экзамен ставится «отлично».

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

Для оценки обучающимися содержания и качества учебного процесса применяется

анкетирование в соответствии с методикой и графиком, утвержденными в установленном 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 с опытом работы по специальности в областях, связанных с реинжинирингом информационных систем, анализом исходного кода и подобных.

**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**

Аудитории для проведения занятий должны быть оснащены проекционной техникой и компьютером с возможностью вывода изображения на проектор.

**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**

Аудитории для проведения занятий должны быть оснащены проекционной техникой и компьютером с возможностью вывода изображения на проектор.

**3.3.4 Характеристики специализированного программного обеспечения**

Специальных требований нет.

**3.3.5 Перечень и объёмы требуемых расходных материалов**

Специальных требований нет.

**3.4. Информационное обеспечение**

**3.4.1 Список обязательной литературы**

1. Оппенгейм А., Шафер Р. Цифровая обработка сигналов. – М.: Техносфера, 2012.

2. Прэтт У. Цифровая обработка изображений. В 2-х кн. – М.: Мир, 1982.

3. Федоров А. Р. Способы кодирования информации для построения программных отказоустойчивых дисковых массивов//Системное программирование. -2012. – Том 7. – Вып.1. – с. 6-31.  
http://sysprog.info/issues\_2012.html - ЭР открытого доступа в сети Интернет

**3.4.2 Список дополнительной литературы**

Не предусмотрено

**3.4.3 Перечень иных информационных источников**

1. <https://software.intel.com/en-us/articles/intel-sdm>

2. <https://www.intel.com/content/www/us/en/processors/carry-less-multiplication-instruction-in-gcm-mode-paper.html>

3. https://www.intel.com/content/www/us/en/architecture-and-technology/advanced-encryption-standard--aes-/data-protection-aes-general-technology.html

**Раздел 4. Разработчики программы**

Кириленко Яков Александрович, старший преподаватель кафедры системного программирования, y.kirilenko@spbu.ru