

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»
(БГТУ им. В.Г. Шухова)



РАБОЧАЯ ПРОГРАММА
дисциплины

Архитектура вычислительных систем

Направление подготовки:
09.03.04 Программная инженерия

профиль подготовки:
Разработка программно-информационных систем

Квалификация (степень)
бакалавр

Форма обучения
очная

Институт информационных технологий и управляющих систем

**Кафедра программного обеспечения вычислительной техники и
автоматизированных систем**

Белгород – 2015

Рабочая программа составлена на основании требований:

- Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.04 «Программная инженерия» (уровень бакалавриата), утверждённого приказом Министерства образования и науки Российской Федерации № 229 от 12 марта 2015 г.
- плана учебного процесса БГТУ им. В.Г. Шухова по направлению подготовки 09.03.04 «Программная инженерия», профиль «Разработка программно-информационных систем».

Составитель: к.ф.-м.н., доцент  (О.В. Осипов)
(ученая степень и звание, подпись) (инициалы, фамилия)

Рабочая программа согласована с выпускающей кафедрой
Программного обеспечения вычислительной техники и автоматизированных систем

Заведующий кафедрой: к.т.н., доцент  (В.М. Поляков)
(ученая степень и звание, подпись) (инициалы, фамилия)

« 16 » 04 2015 г.

Рабочая программа обсуждена на заседании кафедры
Программного обеспечения вычислительной техники и автоматизированных систем

« 16 » 04 2015 г., протокол № 11

Заведующий кафедрой: к.т.н., доцент  (В.М. Поляков)
(ученая степень и звание, подпись) (инициалы, фамилия)

Рабочая программа одобрена методической комиссией института
Информационных технологий и управляющих систем

« 23 » 04 2015 г., протокол № 3/12

Председатель: доцент  (Ю.И. Солопов)
(ученая степень и звание, подпись) (инициалы, фамилия)

1. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Формируемые компетенции			Требования к результатам обучения
№	Код компетенции	Компетенция	
Общепрофессиональные			
1	ОПК-2	владение архитектурой электронных вычислительных машин и систем	<p>В результате освоения дисциплины обучающийся должен</p> <p>Знать:</p> <ul style="list-style-type: none"> – виды и классификацию архитектур современных вычислительных систем; – основные принципы построения вычислительных систем для решения задач различного рода; – принципы функционирования современных многопроцессорных ЭВМ на базе архитектуры x86; – логику работы центрального процессора, сопроцессора, графического процессора, устройств ввода/вывода, внешней и оперативной памяти; – принципы взаимодействия узлов вычислительной системы посредством общей шины, основные виды шин; – основные технические характеристики и показатели быстродействия включённых в состав ЭВМ устройств; <p>Уметь:</p> <ul style="list-style-type: none"> – оценивать технические возможности (быстродействие, производительность и др.) вычислительных систем и их отдельных компонентов; – поддерживать работоспособность вычислительной системы и восстанавливать её после аппаратного сбоя; – собирать вычислительную систему и производить замену её отдельных элементов; <p>Владеть:</p> <ul style="list-style-type: none"> – навыками работы с программным обеспечением для замера производительности и технического состояния вычислительной системы; – навыками настройки и конфигурирования ЭВМ.
Профессиональные			
1	ПК-2	владение навыками использования операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных	<p>В результате освоения дисциплины обучающийся должен</p> <p>Знать:</p> <ul style="list-style-type: none"> – структуру команд центрального процессора и сопроцессора; – виды адресации данных в оперативной памяти, принципы преобразования виртуальных адресов в физические; – принципы оптимизации и отладки программного кода; – структуру программы в оперативной памяти ЭВМ; – способы соединения программного кода на ассемблере с программами, написанными на языках высокого уровня; – структуру и принципы формирования программного кода компиляторами языков структурного и объектно-ориентированного программирования; <p>Уметь:</p> <ul style="list-style-type: none"> – анализировать содержимое произвольного участка

		<p>оперативной памяти, определять местонахождение команд и данных (объектов, переменных, массивов и т.д.);</p> <p>– анализировать машинный код и оценивать его качество с точки зрения оптимальности.</p> <p>Владеть:</p> <p>– навыками разработки программного обеспечения, в том числе dll-библиотек, на ассемблере с использованием пакета masm32;</p> <p>– навыками дизассемблирования, отладки и оптимизации низкоуровневого программного кода с использованием отладчика OllyDbg;</p> <p>– навыками соединения ассемблерного кода с программами на языках C++, C# в среде Microsoft Visual Studio.</p>
--	--	---

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Содержание дисциплины основывается и является логическим продолжением следующих дисциплин:

№	Наименование дисциплины (модуля)
1	Информатика
2	Основы программирования
3	Объектно-ориентированное программирование
4	Алгоритмы и структуры данных
5	Организация ЭВМ и вычислительных систем

Содержание дисциплины служит основой для изучения следующих дисциплин:

№	Наименование дисциплины (модуля)
1	Программирование микроконтроллеров
2	Сети ЭВМ и телекоммуникации
3	Безопасность программно-информационных систем
4	Микропроцессорные системы
5	Параллельное программирование
6	Безопасность программно-информационных систем
7	Администрирование программных и информационных систем

3. ОБЪЁМ ДИСЦИПЛИНЫ

Общая трудоёмкость дисциплины составляет 4 зачётных единицы, 144 часа

Вид учебной работы	Всего часов	Семестр № 5
Общая трудоёмкость дисциплины, час	144	144
Контактная работа (аудиторные занятия), в т.ч.:	68	68
лекции	34	34
лабораторные	34	34
практические	–	–
Самостоятельная работа студентов, в том числе:	76	76
Курсовой проект	–	–
Курсовая работа	–	–
Расчетно-графическое задание	–	–
Индивидуальное домашнее задание	9	9
<i>Другие виды самостоятельной работы</i>	67	67
Форма промежуточной аттестации (зачёт, экзамен)	<i>дифф. зачёт</i>	<i>дифф. зачёт</i>

4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Наименование тем, их содержание и объём

Курс 3 Семестр №5

№ п/п	Наименование раздела (краткое содержание)	Объём на тематический раздел по видам учебной нагрузки, час			
		Лекции	Практические занятия	Лабораторные занятия	Самостоятельная работа
1.	Становление и основные тенденции развития вычислительной техники				
	Исторические этапы развития вычислительной техники. ЭВМ на базе электронно-вакуумных ламп. ЭВМ на базе интегральных схем. Становление и развитие вычислительной техники в СССР и за рубежом. Вычислительная техника будущего. Закон Мура. Технологический процесс. Текущее состояние вычислительной техники. Ведущие производители процессоров и графического оборудования.	4	–	2	6
2.	Виды и классификация вычислительных систем				
	Классификация вычислительных систем по Флинну. Классы параллельных вычислительных систем (SMP, MPP, NUMA). Архитектура с расширенным набором команд. Архитектура с сокращённым набором команд. Архитектура x86. SPARC-архитектура. VLIW-архитектура. Производительность вычислительных систем. Тесты для оценки производительности.	4	–	2	6
3.	Архитектура Джона фон Неймана				
	Основные принципы фон Неймановской архитектуры. Тактовый генератор. Цикл выполнения команды. Представление команд в памяти. Принцип хранимой в памяти программы.	4	–	2	4
4.	Устройство процессора				
	Характеристики процессора. Арифметико-логическое устройство. Устройство управления. Регистры и флаги процессора. Арифметические и логические команды процессора для работы с целочисленной арифметикой. Команды передачи управления. Команды для работы со стеком. Цепочечные команды. Команды пересылки данных. 64-разрядная архитектура.	4	–	10	20
5.	Организация шин				
	Характеристики и виды шин. Последовательные и параллельные шины. Централизованный и децентрализованный арбитраж шин. Алгоритмы динамического изменения приоритетов при организации арбитража. Системы ввода/вывода.	4	–	4	8
6.	Память				
	Иерархия памяти компьютера. Оперативная память.	6	–	4	8

	Кэш-память. Характеристики и виды оперативной памяти. Сплошная и сегментированная модели памяти. Внешняя память компьютера. Жёсткий диск. Твердотельный накопитель. Кластеризация.				
7.	Программирование на ассемблере				
	Структура ассемблерной программы. Компиляция и отладка ассемблерных программ с использованием пакета <code>masm32</code> и отладчика <code>OllyDbg</code> . Представление в сегменте данных чисел, строк, массивов. Директивы объявления данных. Назначение стека. Вызов подпрограмм. Устройство сопроцессора. Стили вызова <code>stdcall</code> , <code>cdecl</code> , <code>fastcall</code> , <code>thiscall</code> , <code>pascal</code> . Разработка <code>dll</code> -библиотек на ассемблере. Статический и динамический способы подключения <code>dll</code> -библиотек.	8	–	10	24
	ВСЕГО	34		34	76

4.2. Содержание лабораторных занятий

№ п/п	Наименование раздела дисциплины	Тема лабораторного занятия	Кол-во лаб. часов	К-во часов СРС
семестр №5				
1	Становление и основные тенденции развития вычислительной техники	Разработка программ на ассемблере. Работа с отладчиком <code>OllyDbg</code> и пакетом <code>masm32</code> . Структура программы: сегмент данных, сегмент кода, сегмент стека. Директивы объявления данных. Представление переменных и массивов в сегменте данных. Трассировка программы.	4	8
2	Архитектура Джона фон Неймана, Устройство процессора	Структура команд процессора. Код операции. Представление команд в памяти ЭВМ. Режимы адресации. Построение машинного кода команды по её названию.	6	10
3	Устройство процессора	Арифметические команды центрального процессора. Команды процессора для выполнения сложения, вычитания, умножения и деления целых чисел. Представление в памяти ЭВМ целых знаковых и беззнаковых чисел. Команды изменения размерности и знака числа. Команды пересылки данных.	2	8
4	Организация шин, Программирование на ассемблере	Команды передачи управления. Условный и безусловный переход. Команды для организации циклов. Флаги процессора. Вызов подпрограмм. Соглашение о вызове подпрограмм. Консольный ввод и вывод. Назначение стека. Обработка массивов.	6	6
5	Программирование на ассемблере	Команды сопроцессора для работы с целыми и вещественными числами. Регистры сопроцессора. Алгоритм возведения числа x в степень y с использованием команд сопроцессора.	2	5
6	Программирование	Логические команды и команды сдвига	2	5

	на ассемблере	центрального процессора для работы с целыми числами.		
7	Программирование на ассемблере	Цепочечные команды центрального процессора. Обработка массивов и строк с использованием цепочечных команд.	2	5
8	Виды и классификация вычислительных систем, Память	Способы вызова ассемблерных подпрограмм в языках высокого уровня. Стили вызова подпрограмм fastcall, stdcall, pascal, cdecl. Ассемблерная вставка. Разработка dll-библиотек на ассемблере. Сравнение производительности кода, написанного на ассемблере и кода, полученного компиляторами C++, C#, Pascal.	6	5
9	Память	Принципы формирования объектно-ориентированного кода. Стилль вызова thiscall. Таблица виртуальных методов.	4	8
ИТОГО:			34	60

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

5.1. Перечень контрольных вопросов (типовых заданий)

№ п/п	Наименование раздела дисциплины	Содержание вопросов (типовых заданий)
1	Становление и основные тенденции развития вычислительной техники	Этапы развития вычислительной техники. ЭВМ первого поколения на базе электронно-вакуумных ламп. ЭВМ на базе транзисторов и интегральных схем. Микропроцессорные ЭВМ. Характеристики современных вычислительных систем. Закон Мура. Технологический процесс. Перспективы развития вычислительной техники и систем искусственного интеллекта.
2	Виды и классификация вычислительных систем	Классификация вычислительных систем по Флинну. Единицы для измерения производительности вычислительных систем. Пакеты программ для оценки производительности. Преимущества VLIW-архитектуры по сравнению с x86. Различия между CISC и RISC-системами. Для решения какого класса задач больше подходят системы с параллельной архитектурой? Отличия 64-разрядной и 32-разрядной архитектур. Распараллеливание вычислений с помощью графических ускорителей.
3	Архитектура Джона фон Неймана	Структурная схема классической фон Неймановской архитектуры. Основные принципы архитектуры Джона фон Неймана. Цикл выполнения команды. Назначение устройства управления и АЛУ. Порядок выборки команд из памяти. Какую информацию содержат поля d , w , r/m , mod в коде команды. В каком случае в коде команды присутствует байт SIB и какая в нём содержится информация?
4	Устройство процессора	Структура регистров центрального процессора. Назначение регистров ESP и EIP. Принцип работы команд CALL и RET. Использование указателя EBP для фиксирования адреса в стеке внутри подпрограммы. Назначение индексных регистров. Назначение флагов CF, AF, OF, ZF, SF. Написать

		последовательность команд, моделирующую выполнение команд PUSH, POP, PUSHAD, POPAD. Отличие команд DIV и IDIV. Почему команды SUB, ADD работают и со знаковыми и с беззнаковыми числами? Каким образом ЦП расширяет целые числа? Отличие знаковой и беззнаковой арифметики. Работа команд CMP, LOOP.
5	Организация шин	Характеристики шин USB, SATA, AGP, PCI-Express. Централизованный и децентрализованный арбитраж шин. Гнездо центрального процессора.
6	Память	<p>Способы адресации операндов в памяти. Явная и неявная адресация операндов. Назначение стека. Выделение памяти для локальных переменных и массивов. Отличие способов выделения памяти для глобальных и локальных переменных. Очистка стека. Чем в техническом смысле отличается передача аргумента в виде <code>void (t_item a)</code>, <code>void (t_item* a)</code> и <code>void (t_item& a)</code>? Куда помещается возвращаемое функцией значение.</p> <p>Задачи по программированию:</p> <ol style="list-style-type: none"> 1. Определить местонахождение переменных и массивов, объявленных в программе, в сегменте данных. 2. Написать программу на ассемблере для заполнения массива длиной n кубами чисел $1^3, 2^3, 3^3, \dots, n^3$. 3. Написать на ассемблере подпрограмму <code>int eval(int* a, int* b, int n)</code>, которая возвращает значение выражения: $A = \sum_{i=1}^n \frac{a_i}{5} + ib_i$ 4. Написать на ассемблере подпрограмму <code>void output(double* a, int n)</code> для вывода на экран массива вещественных чисел в виде: <code>a[1] = ...; a[2] = ...;...</code> 5. С использованием команд сопроцессора написать подпрограмму <code>res(int n)</code>, возвращающую значение выражения: $R = \sum_{k=1}^n \left(\sin^2\left(\frac{1}{k}\right) + \left(\cos^2\left(\frac{1}{2k}\right) - 1 \right) \right)$
7	Программирование на ассемблере	<p>Соглашения о вызовах. Декорирование названий подпрограмм. Стили вызова <code>stdcall</code>, <code>cdecl</code>, <code>fastcall</code>, <code>pascal</code>. Отличие динамического и статического способа подключения <code>dll</code>-библиотеки.</p> <ol style="list-style-type: none"> 1. Написать подпрограмму <code>count</code> в стиле <code>fastcall</code>, которая возвращает количество цифр в восьмеричном представлении числа n. В основной программе ввести число n с клавиатуры и вывести результат. Глобальные переменные в подпрограмме <code>count</code> использовать не разрешается. <pre>data n dd ? .code count proc ... count endp start: ... ; Ввод числа n с клавиатуры ... ; Передать число n в качестве аргумента подпрограмме </pre>

	<pre> ... call count ; Вывод результата 2. Написать на ассемблере подпрограмму в стиле thiscall для сложения и произведения комплексных чисел и подключить её к классу на языке C++. class complex { Double Re, Im; complex operator + (const complex& a); complex operator * (const complex& a); }; </pre>
--	---

5.2. Перечень тем курсовых проектов, курсовых работ, их краткое содержание и объём.

Курсовые работы и курсовые проекты по дисциплине «Архитектура вычислительных систем» учебным планом не предусмотрены.

5.3. Перечень индивидуальных домашних заданий, расчетно-графических заданий.

Учебным планом предусмотрено одно индивидуальное домашнее задание, для выполнения которого предусмотрено 9 часов самостоятельной работы студента.

Цель ИДЗ: разработка оконного графического Windows-приложения на ассемблере с использованием пакета masm32 и отладчика OllyDbg. Разрабатываемая программа должна решать задачу анимации процесса движения графических примитивов внутри окна с использованием API-функций ОС Windows для работы с GDI-графикой.

Типовые задания ИДЗ:

1. Создать анимацию движения абсолютно упругих мячей в невесомости, которые взаимодействуют с границей окна, с другими графическими примитивами и друг с другом.
2. Создать анимацию движения упругих мячей в поле силы тяжести, параметры которого задаются в программе.

ИДЗ включает в себя следующие разделы: цель задачи, описание программы в виде блок-схем, исходный код программы, результаты выполнения программы. Оценка ИДЗ производится по результатам проверки пояснительной записки и работоспособности написанной программы, а также по результатам защиты, которая проходит в виде устной беседы с преподавателем.

5.4. Перечень контрольных работ

Контрольные работы по дисциплине «Архитектура вычислительных систем» учебным планом не предусмотрены.

6. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

6.1. Перечень основной литературы

1. Цилькер Б.Я., Орлов С.А. Организация ЭВМ и систем: Учебник для вузов. – СПб.: Питер, 2006. – 686 с.: ил.
2. Юров В.И. Assembler. Учебник для вузов. 2-е изд. – СПб.: Питер, 2006.
3. Организация ЭВМ и систем. Основы программирования на языке Ассемблер: методические указания к выполнению лабораторных работ для студентов направлений бакалавриата 230100 – Информатика и вычислительная техника, 231000 – Программная инженерия и специальности 090303 – Информационная безопасность автоматизированных систем / сост.: А.И. Гарибов, О.В. Осипов. – Белгород: Изд-во БГТУ, 2014. – 35 с.
4. Мищенко В.К. Архитектура высокопроизводительных вычислительных систем [Электронный ресурс]: учебное пособие/ Мищенко В.К.– Электрон. текстовые данные. – Новосибирск: Новосибирский государственный технический университет, 2013. – 40 с. – Режим доступа: <http://www.iprbookshop.ru/44898>.– ЭБС «IPRbooks».
5. Чекмарев Ю.В. Вычислительные системы, сети и телекоммуникации [Электронный ресурс]/ Чекмарев Ю.В.– Электрон. текстовые данные. – М.: ДМК Пресс, 2013.– 184 с.– Режим доступа: <http://www.iprbookshop.ru/5083>.– ЭБС «IPRbooks».
6. Шандаров, Е.С. Архитектура вычислительных систем. Компьютерный лабораторный практикум. [Электронный ресурс]: Учебные пособия – Электрон. дан. – М. : ТУСУР, 2012. – 44 с. – Режим доступа: <http://e.lanbook.com/book/11261>.
7. Осипов О.В. Организация ЭВМ и вычислительных систем: методические указания к выполнению лабораторных работ для студентов специальности 090303.65 – Информационная безопасность автоматизированных систем / сост.: О.В. Осипов. – Белгород: Изд-во БГТУ, 2015. – 115 с.

6.2. Перечень дополнительной литературы

1. Каган Б. М. Электронные вычислительные машины и системы. М.: Энергоатомиздат, 1991.
2. Таненбаум, Э. Архитектура компьютера / Э. Таненбаум. – 4-е изд. – СПб.: Питер, 2003. – 698 с. – (Классика computer science). – ISBN 5-318-00298-6.
3. Пирогов, В. Ю. Ассемблер и дизассемблирование / В. Ю. Пирогов. – СПб.: БХВ-Петербург, 2006. – 447 с. + 1 эл. опт. диск (CD-ROM). – ISBN 5-94157-677-3.
4. Дашевский Л. Н., Шкабара Е. А. Как это начиналось (Воспоминания о создании первой отечественной электронной вычислительной машины – МЭСМ). – М.: «Знание», 1981. – 64 с. (Новое в жизни, науке, технике. Сер. «Математика, кибернетика», № 1).

Справочная и нормативная литература

1. ГОСТ 27201-87 Машины вычислительные электронные персональные. Типы, основные параметры, общие технические требования.
2. ГОСТ 2.708-81 Единая система конструкторской документации. Правила выполнения электрических схем цифровой вычислительной техники.
3. ГОСТ 25123-82 Машины вычислительные и системы обработки данных. Техническое задание. Порядок построения, изложения и оформления.
4. ГОСТ Р МЭК 62623-2015 Компьютеры настольные и ноутбуки. Измерение потребления энергии.
5. ГОСТ 25861-83 Машины вычислительные и системы обработки данных. Требования по электрической и механической безопасности и методы испытаний.
6. ГОСТ 24750-81 Средства технические вычислительной техники. Общие требования технической эстетики.
7. ГОСТ 23336-78 Машины вычислительные аналоговые и аналого-цифровые. Правила

выполнения схем моделирования.

6.3. Перечень интернет ресурсов

1. <http://prog-cpp.ru/asm/>
2. <http://www.club155.ru/x86cmd>
3. <http://asmworld.ru/>
4. <http://www.i-assembler.ru/>
5. https://ru.wikipedia.org/wiki/Соглашение_о_вызове
6. <http://natalia.appmat.ru/c&c++/dll.html>

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

Для проведения лекционных занятий требуется аудитория с доской (ГК426, ГК 430), комплектом пишущих маркеров для досок, компьютером и проектором.

Для проведения лабораторных занятий необходим компьютерный зал, в котором имеется следующее программное обеспечение:

Microsoft Windows;

Microsoft Office;

Среды программирования Free Pascal, Dev C++ или CodeBlocks (Свободно распространяемое ПО);

Пакет для компиляции ассемблерных программ `masm32` (Свободно распространяемое ПО);

Отладчик 32-разрядных приложений OllyDbg (Свободно распространяемое ПО).

ПРИЛОЖЕНИЯ

Приложение №1. Методические указания для обучающегося по освоению дисциплины.

Дисциплина «Архитектура вычислительных систем» входит в базовый блок учебного плана специальности 09.03.04 «Программная инженерия». Для успешного освоения курса требуются знания основ программирования и информатики.

Целью курса является изучение архитектур вычислительных систем различного типа, тенденции развития вычислительной техники, устройство современных ЭВМ и их отдельных компонентов. На лекционных занятиях детально рассматривается архитектура x86 (устройство процессора, сопроцессора, механизмы адресации памяти), работа оперативной памяти и кэш-памяти, систем ввода-вывода, графического процессора, внешних устройств. На лабораторных занятиях обучающиеся осваивают программирование на ассемблере с использованием пакета `masm32`, среды Microsoft Visual Studio, отладчика OllyDbg. Также рассматриваются вопросы подключения ассемблерных подпрограмм к программам на языках высокого уровня с помощью dll-библиотек.

Методические указания к выполнению лабораторных работ даны в пособии:

Осипов О.В. Организация ЭВМ и вычислительных систем: методические указания к выполнению лабораторных работ для студентов специальности 090303.65 – Информационная безопасность автоматизированных систем / сост.: О.В. Осипов. – Белгород: Изд-во БГТУ, 2015. – 115 с.
--

Для выполнения лабораторных и самостоятельных работ обучающийся использует лекционный материал, а также литературу, перечисленную в списке основной литературы пункта б.1.

Учебный план предусматривает проведение лабораторных и лекционных занятий, выполнение одного индивидуального домашнего задания. Защита лабораторных работ проходит в виде беседы с преподавателем и выполнения задач по программированию. Итоговый контроль знаний осуществляется в форме зачёта с оценкой.

Основной целью курса является выработка у обучающихся понимания работы программ на уровне команд процессора, получения навыков дизассемблирования, анализа и оптимизации машинного кода.

6. ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

6.1. Перечень основной литературы

1. Цилькер Б.Я., Орлов С.А. Организация ЭВМ и систем: Учебник для вузов. – СПб.: Питер, 2006. – 686 с.: ил.
2. Юров В.И. Assembler. Учебник для вузов. 2-е изд. – СПб.: Питер, 2006.
3. Организация ЭВМ и систем. Основы программирования на языке Ассемблер: методические указания к выполнению лабораторных работ для студентов направлений бакалавриата 230100 – Информатика и вычислительная техника, 231000 – Программная инженерия и специальности 090303 – Информационная безопасность автоматизированных систем / сост.: А.И. Гарибов, О.В. Осипов. – Белгород: Изд-во БГТУ, 2014. – 35 с.
4. Мищенко В.К. Архитектура высокопроизводительных вычислительных систем [Электронный ресурс]: учебное пособие/ Мищенко В.К.– Электрон. текстовые данные. – Новосибирск: Новосибирский государственный технический университет, 2013. – 40 с. – Режим доступа: <http://www.iprbookshop.ru/44898>
5. Богданов А.В. Архитектуры и топологии многопроцессорных вычислительных систем [Электронный ресурс]/ А.В. Богданов [и др.]. – Электрон. текстовые данные. – М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. – 135 с.– Режим доступа: <http://www.iprbookshop.ru/52189>
6. Чекмарев Ю.В. Вычислительные системы, сети и телекоммуникации [Электронный ресурс] / Ю.В. Чекмарев. — Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 184 с. — 978-5-4488-0071-9. — Режим доступа: <http://www.iprbookshop.ru/63576.html>
7. Осипов О.В. Организация ЭВМ и вычислительных систем: методические указания к выполнению лабораторных работ для студентов специальности 090303.65 – Информационная безопасность автоматизированных систем / сост.: О.В. Осипов. – Белгород: Изд-во БГТУ, 2015. – 115 с.
8. Лиманова Н.И. Архитектура вычислительных систем и компьютерных сетей [Электронный ресурс]: учебное пособие / Н.И. Лиманова. — Самара: Поволжский государственный университет телекоммуникаций и информатики, 2017. — 197 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/75368.html>

6.2. Перечень дополнительной литературы

1. Каган Б. М. Электронные вычислительные машины и системы. М.: Энергоатомиздат, 1991.
2. Таненбаум, Э. Архитектура компьютера / Э. Таненбаум. – 4-е изд. – СПб.: Питер, 2003. – 698 с. – (Классика computer science). – ISBN 5-318-00298-6.
3. Пирогов, В. Ю. Ассемблер и дизассемблирование / В. Ю. Пирогов. – СПб.: БХВ-Петербург, 2006. – 447 с. + 1 эл. опт. диск (CD-ROM). – ISBN 5-94157-677-3.
3. Дашевский Л. Н., Шкабара Е. А. Как это начиналось (Воспоминания о создании первой отечественной электронной вычислительной машины – МЭСМ). – М.: «Знание», 1981. – 64 с. (Новое в жизни, науке, технике. Сер.

«Математика, кибернетика», № 1).

4. Крахоткина Е.В. Архитектура ЭВМ [Электронный ресурс]: учебное пособие (лабораторный практикум) / Е.В. Крахоткина, В.И. Терехин. — Ставрополь: Северо-Кавказский федеральный университет, 2015. — 80 с. — Режим доступа: <http://www.iprbookshop.ru/63074.html>

**Рабочая программа и ГРС без изменений утверждена
на 2016 / 2017 учебный год**

Протокол № 10 заседания кафедры от « 9 » 06 2016 г.

Заведующий кафедрой _____
(подпись, Ф.И.О.)

Директор института _____
(подпись, Ф.И.О.)

**Рабочая программа и ГРС без изменений утверждена
на 2017 / 2018 учебный год**

Протокол № 11 заседания кафедры от « 22 » 05 2017 г.

Заведующий кафедрой _____
(подпись, Ф.И.О.)

Директор института _____
(подпись, Ф.И.О.)

**Рабочая программа и ГРС с изменениями,
дополнениями утверждена на 2018 / 2019 учебный год**

Протокол № 10 заседания кафедры от « 21 » 05 2018 г.

Заведующий кафедрой _____
(подпись, Ф.И.О.)

Директор института _____
(подпись, Ф.И.О.)

8. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ

Утверждение рабочей программы без изменений
Рабочая программа без изменений утверждена на 2019/2020 учебный
год.

Протокол № 10 заседания кафедры от «18» мая 2019 г.

Заведующий кафедрой _____ В.М. Поляков
подпись, ФИО

Директор института _____ А.В. Белоусов

7. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ³

Рабочая программа утверждена на 20 20 /20 21 учебный год
без изменений / с изменениями, дополнениями⁴

Протокол № 8 заседания кафедры от « 21 » 04 20 20 г.

Заведующий кафедрой _____ (Поляков В.М.)
подпись, ФИО

Директор института _____ (Белоусов А.В.)
подпись, ФИО

³ Заполняется каждый учебный год на отдельных листах

⁴ Нужно подчеркнуть

7. УТВЕРЖДЕНИЕ РАБОЧЕЙ ПРОГРАММЫ

Рабочая программа утверждена на 2021/2022 учебный год
без изменений²

Протокол № 8 заседания кафедры от « 15 » мая 2021 г.

Заведующий кафедрой _____

подпись, ФИО

Полков В.М.

Директор института _____

подпись, ФИО

Белоусов А.В.

¹ Заполняется каждый учебный год на отдельных листах

² Нужно подчеркнуть