

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ОБНИНСКИЙ ИНСТИТУТ АТОМНОЙ ЭНЕРГЕТИКИ  
– филиал федерального государственного автономного образовательного  
учреждения высшего образования  
«Национальный исследовательский ядерный университет «МИФИ»  
(ИАТЭ НИЯУ МИФИ)

**ТЕХНИКУМ ИАТЭ НИЯУ МИФИ**



**УТВЕРЖДАЮ:**

И.о. зам. директора по учебной деятельности  
ИАТЭ НИЯУ МИФИ

М.Г.Ткаченко

«31» 08 2018 г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

ОП.18 Системное программирование  
*название дисциплины*

по специальности среднего профессионального образования

**09.02.04 Информационные системы (по отраслям)**  
*код, наименование специальности*

Форма обучения  
*очная*

Обнинск 2018

Рабочая программа дисциплины разработана на основе Федерального государственного образовательного стандарта среднего профессионального образования по специальности **09.02.04 Информационные системы (по отраслям)**, утверждённого приказом Министерства образования и науки Российской Федерации от 14 мая 2014 г. № 525.

**Программу составил:**

Константинов Роман Юрьевич, преподаватель высшей квалификационной категории

**Рецензенты:**

Розман Л.С. – Генеральный директор ООО «Персона»

Саркисова С.О. – преподаватель высшей квалификационной категории техникума ИАТЭ НИЯУ МИФИ.

Программа рассмотрена на заседании предметно-цикловой комиссии по специальностям


09.02.04 Информационные системы (по отраслям) и

09.02.03 Программирование в компьютерных системах

Протокол № 1/18-19 от 30.08.2018

Программа рассмотрена на заседании методического совета Техникума  
Протокол № 1 от 30.08.2018

Председатель ПЦК

 С.О.Саркисова

«30» августа 2018 г.

Председатель методического совета

 С.А.Косарев

«30» августа 2018 г.

Составитель программы

 Р.Ю.Константинов

«30» августа 2018 г.

# **1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**

## **«Системное программирование»**

### **1.1. Область применения программы**

Программа учебной дисциплины (далее программа) – является частью примерной основной профессиональной образовательной программы в соответствии с ФГОС по специальностям СПО **09.02.04. Информационные системы(по отраслям)** (базовая подготовка) в части освоения основного вида профессиональной деятельности (ВПД): и соответствующих профессиональных компетенций (ПК):

ПК 1.1. Выполнять разработку спецификаций отдельных компонент.

ПК 1.2. Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.

ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4. Выполнять тестирование программных модулей.

ПК 1.5. Осуществлять оптимизацию программного кода модуля.

ПК 1.6. Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.

### **1.2. Цели и задачи дисциплины**

С целью овладения указанным видом профессиональной деятельности и соответствующими профессиональными компетенциями обучающийся в ходе освоения профессионального модуля должен:

**иметь практический опыт:**

- разработки алгоритма поставленной задачи и реализации его средствами автоматизированного проектирования;
- разработки кода программного продукта на основе готовой спецификации на уровне модуля;
- использования инструментальных средств на этапе отладки программного продукта;
- проведения тестирования программного модуля по определенному сценарию;

**уметь:**

- осуществлять разработку кода программного модуля на современных языках программирования;
- создавать программу по разработанному алгоритму как отдельный модуль;
- выполнять отладку и тестирование программы на уровне модуля;
- оформлять документацию на программные средства;
- использовать инструментальные средства для автоматизации оформления документации;

**знать:**

- основные этапы разработки программного обеспечения;
- основные принципы технологии структурного и объектно-ориентированного программирования;
- основные принципы отладки и тестирования программных продуктов;
- методы и средства разработки технической документации.

### **1.3. Рекомендуемое количество часов на освоение программы учебной дисциплины:**

максимальной учебной нагрузки обучающегося – 76 часов, включая:

обязательной аудиторной учебной нагрузки обучающегося – 45 часов;

самостоятельной работы обучающегося – 31 часов;

## 2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результатом освоения программы учебной дисциплины является овладение обучающимися видом профессиональной деятельности, в том числе профессиональными (ПК) и общими (ОК) компетенциями:

Код	Наименование результата обучения
ПК 1.1	Выполнять разработку спецификаций отдельных компонент.
ПК 1.2	Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.
ПК 1.3	Выполнять отладку программных модулей с использованием специализированных программных средств.
ПК 1.4	Выполнять тестирование программных модулей.
ПК 1.5	Осуществлять оптимизацию программного кода модуля.
ПК 1.6	Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.
ОК 1	Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.
ОК 2	Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.
ОК 3	Решать проблемы, оценивать риски и принимать решения в нестандартных ситуациях.
ОК 4	Осуществлять поиск, анализ и оценку информации, необходимой для постановки и решения профессиональных задач, профессионального и личностного развития.
ОК 5	Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.
ОК 6	Работать в коллективе и команде, обеспечивать ее сплочение, эффективно общаться с коллегами, руководством, потребителями.
ОК 7	Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий.
ОК 8	Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.
ОК 9	Быть готовым к смене технологий в профессиональной деятельности.

### 3. СТРУКТУРА И ПРИМЕРНОЕ СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

#### 3.1. Тематический план учебной дисциплины

Коды профессиональных компетенций	Наименования разделов профессионального модуля	Всего часов (макс. учебная нагрузка и практики)	Объем времени, отведенный на освоение междисциплинарного курса (курсов)					Практика	
			Обязательная аудиторная учебная нагрузка обучающегося			Самостоятельная работа обучающегося		Учебная, часов	Производственная (по профилю специальности), часов
			Всего, часов	в т.ч. лабораторные работы и практические занятия, часов	в т.ч., курсовая работа (проект), часов	Всего, часов	в т.ч., курсовая работа (проект), часов		
1	2	3	4	5	6	7	8	9	10
ПК 1.2.	Раздел 1. Разработка кода программного продукта на основе готовых спецификаций на уровне модуля	39	26	-	-	13	-	-	-
ПК 1.3. -1.5.	Раздел 2. Отладка, тестирование и оптимизация программных модулей с использованием специализированных программных средств.	60	40	34	-	20	-	-	-
ПК 1.6.	Раздел 3. Разработка компонентов проектной и технической документации с использованием графических языков спецификаций.	57	38		30	19	15	-	-
	<b>Итого по модулю</b>	<b>618</b>	<b>244</b>	108	30	<b>122</b>		<b>108</b>	<b>144</b>

### 3.2. Содержание обучения по учебной дисциплине

Наименование разделов профессионального модуля (ПМ), междисциплинарных курсов (МДК) и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся	Объем часов	Требования к практическому опыту, знаниям, умениям	Уровень освоения
1	2	3	4	5
<b>Раздел 1. Разработка кода программного продукта на основе готовых спецификаций на уровне модуля</b>		<b>30</b>		
<b>Тема 1.1. Основные понятия и определения</b>	<b>Содержание</b> (указывается перечень дидактических единиц)	<b>2</b>	ОК 1-10 ПК 1.1	2
	1. Программы и программное обеспечение Системное программирование. Этапы подготовки программы	2		
<b>Тема 1.2. Ассемблеры</b>	<b>Содержание</b> (указывается перечень дидактических единиц)	<b>4</b>	ОК 1-10 ПК 1.1	2
	1. Программирование на языке Ассемблера. Предложения языка Ассемблера. Регистры. Биты и байты ASCII. Двоичные числа. Шестнадцатеричное представление	2		
	2. Сегменты. Расширение набора команд. Способы адресации. Директивы Директивы определения данных. Директива определения байта (DB) Директива определения слова (DW)	2		
<b>Тема 1.3. Регистры</b>	<b>Содержание</b> (указывается перечень дидактических единиц)	<b>2</b>	ОК 1-10 ПК 1.1	2
	1. Сегментные регистры: CS, DS, SS и ES Регистры общего назначения: AX, BX, CX и DX Регистровые указатели: SP и BP Индексные регистры: SI и DI Регистр командного указателя: IP Флаговый регистр	2		
<b>Тема 1.4. Арифметические операции</b>	<b>Содержание</b> (указывается перечень дидактических единиц)	<b>2</b>	ОК 1-10 ПК 1.1	2
	1. Обработка двоичных данных Беззнаковые и знаковые данные Умножение Сдвиг регистровой пары DX:AX Деление Преобразование знака Обработка данных в форматах ASCII и BCD Двоично-десятичный формат (BCD) Сдвиг и округление	2		
<b>Тема 1.5. Команды обработки строк. Обработка таблиц</b>	<b>Содержание</b> (указывается перечень дидактических единиц)	<b>4</b>	ОК 1-10 ПК 1.1	2
	1. Свойства операций над строками REP: Префикс повторения цепочечной команды. MOVS: Пересылка строк. LODS: Загрузка	2		

		строки. STOS: Запись строки. CMPS: Сравнение строк. SCAS: Сканирование строк. Сканирование и замена. Альтернативное кодирование. Дублирование образца			
	2.	Определение таблиц. Прямой табличный доступ. Табличный поиск. Транслирующая команда XLAT. Операторы типа, длина и размеры.	2		
<b>Тема 1.6. Команды обработки строк</b>	<b>Содержание</b> (указывается перечень дидактических единиц)		<b>4</b>	ОК 1-10 ПК 1.1	2
	1	Свойства операций над строками. REP: Префикс повторения цепочечной команды. MOVS: Пересылка строк. LODS: Загрузка строки. STOS: Запись строки	2		
	2	CMPS: Сравнение строк. SCAS: Сканирование строк. Сканирование и замена. Альтернативное кодирование. Дублирование образца.	2		
<b>Тема 1.7. Обработка таблиц</b>	<b>Содержание</b> (указывается перечень дидактических единиц)		<b>2</b>	ОК 1-10 ПК 1.1	2
	1	Определение таблиц. Прямой табличный доступ. Табличный поиск. Транслирующая команда XLAT. Операторы типа, длина и размеры.	2		
<b>Тема 1.8. Требования языка</b>	<b>Содержание</b> (указывается перечень дидактических единиц)		<b>2</b>	ОК 1-10 ПК 1.1	2
	1	Комментарии в программах на Ассемблере. Формат кодирования. Директивы. Память и регистры. Инициализация программы.	2		
<b>Тема 1.9. Ввод и выполнение программ</b>	<b>Содержание</b> (указывается перечень дидактических единиц)		<b>2</b>	ОК 1-10 ПК 1.1	2
	1	Ввод программы. Подготовка программы для выполнения. Ассемблирование программы. Компоновка программы. Выполнение программы. Файл перекрестных ссылок.	2		
<b>Тема 1.10. Алгоритмы работы Ассемблеров</b>	<b>Содержание</b> (указывается перечень дидактических единиц)		<b>2</b>	ОК 1-10 ПК 1.1	2
	1	Двухпроходный Ассемблер — первый проход. Структура таблиц Ассемблера Двухпроходный Ассемблер — второй проход. Некоторые дополнительные директивы. Директивы связывания. Одно и многопроходный Ассемблер.	2		
<b>Самостоятельная работа при изучении раздела ПМ 4.</b> Работа над рефератом по предложенным темам и составление отчета к решаемым задачам: <ol style="list-style-type: none"> <li>1. Место языков ассемблера среди языков программирования.</li> <li>2. Структура МП Intel 80x86: используемые регистры.</li> <li>3. Структура МП Intel 80x86: операционное устройство и шинный интерфейс.</li> <li>4. Размещение данных в памяти. Сегментация памяти.</li> <li>5. Структура регистра флагов. Команды установки флагов.</li> <li>6. Структура и форматы команд МП Intel 80x86. Команды пересылки данных.</li> <li>7. Способы адресации в командах МП Intel 80x86.</li> </ol>			<b>13</b>		
<b>Раздел 2. Отладка, тестирование и оптимизация программных модулей</b>			<b>15</b>		
<b>Тема 2.1. Отладка, тестирование и оптимизация программ на языке ассемблера</b>	<b>Содержание</b> (указывается перечень дидактических единиц)		<b>6</b>	ОК 1-10 ПК 1.2	
	1.	<b>Модульное программирование.</b> Одномодульное программирование: использование	2		2

		функций, классификация параметров, составление функций с переменным списком параметров. Составление рекурсивных функций на ассемблере.			
	2.	Многомодульное программирование: использование внешних функций, использование локальных областей памяти. Стыковка с языками высокого уровня. Особенности создания ассемблерных функций для включения их в dll. Макросредства языка ассемблера. Ввод-вывод данных. Работа с файлами	2		
	3.	<b>Отладка, тестирование и оптимизация программ на языке ассемблера</b> Просмотр и исполнение кода программы. Внесение в тело программы точек выхода, которые останавливают исполнение программы. Проверка и изменение в памяти значений переменных. Ассемблирование и реассемблирование кода.	2		
		<b>Практические работы</b>	<b>30</b>		
	1.	Этапы разработки программы на ассемблере	2		3
	2.	Трансляция, компоновка, и отладка программ	2		
	3.	Режимы адресации	2		
	4.	Десятичные числа. Сложение и вычитание целых чисел Умножение и деление целых чисел.	2		
	5.	Программирование арифметических задач	2		
	6.	Логические команды и команды сдвига	2		
	7.	Программирование ветвлений и циклов Команды условного перехода. Команда безусловного перехода. Организация циклов	2		
	8.	Применение команд цикла для программирования действий с массивами данных. Организация циклов	2		
	9.	Массивы	2		
	10.	Команды пересылки данных. стек	2		
	11.	Программирование операций ввода-вывода	2		
	12.	Подпрограммы	2		
	13.	Работа с файлами	2		
	14.	Работа с графикой	2		
	15.	Комплексная работа	2		
	<b>Самостоятельная работа при изучении раздела 2.</b> Работа над рефератом по предложенным темам и составление отчета к решаемым задачам:		<b>20</b>		
	1. Место языков ассемблера среди языков программирования.				
	2. Структура МП Intel 80x86: используемые регистры.				
	3. Структура МП Intel 80x86: операционное устройство и шинный интерфейс.				
	4. Размещение данных в памяти. Сегментация памяти.				
	5. Структура регистра флагов. Команды установки флагов.				
	6. Структура и форматы команд МП Intel 80x86. Команды пересылки данных.				
	7. Способы адресации в командах МП Intel 80x86.				
	8. Представление данных в IBM PC: целые числа.				
	9. Представление данных в IBM PC: двоично-десятичные числа.				
	10. Представление данных в IBM PC: алфавитно-цифровые данные.				



## **4. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**

### **4.1. Требования к минимальному материально-техническому обеспечению**

Реализация программы модуля предполагает наличие учебных кабинетов информатики и информационных технологий; лабораторий информатики и вычислительной техники.

Оборудование учебного кабинета и рабочих мест кабинета информатики и вычислительной техники: рабочие места по количеству обучающихся, компьютеры на рабочем месте учащихся с лицензионным программным обеспечением.

Технические средства обучения: персональные компьютеры с лицензионным программным обеспечением, мультимедийная система

Программное обеспечение:

- Операционная система Windows XP, 7;
- Язык программирования Assembler;
- пакет прикладных программ Microsoft Office;

### **4.2. Информационное обеспечение обучения**

**Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы**

Основные источники:

1. Assembler. : Практикум. 2-е изд./ В. И. Юров. - СПб : Питер, 2006 г.
2. Assembler для DOS, Windows и UNIX - 3-е изд., стер./ С.В. Зубков. - СПб : Питер, 2004 г.
3. Assembler: Учебник для вузов / В.И. Юров. - СПб : Питер, 2010 г.
6. Ассемблер для Windows: 3-е изд., перераб. и доп./ В.Ю. Пирогов. - СПб. : БХВ-Петербург, 2005 г.
7. Искусство программирования на Ассемблере./ Н. Г. Голубь. - СПб : "ДиаСофт Юп", 2006 г.
8. В.Юров, С. Хорошенко Assembler: учебный курс Питер 2004 г.
9. Зубков С.В. Assembler для DOS, Windows и Unix ДМК 2004 г.
10. Юров В. И. Assembler. Специальный справочник Питер 2006 г.

Дополнительные источники:

1. Использование Turbo Assembler при разработке программ. Киев, Диалектика, 2003.
2. П.И. Рудаков, К.Г. Финогенов. Программируем на языке ассемблера IBM PC. Обнинск, 2005.
3. В.Н. Пильщиков. Программирование на языке ассемблера IBM PC. М. Диалог-МИФИ, 2005
4. Том Сван. Освоение Turbo Assembler. II издание. Киев, Диалектика, 2002.
5. А.В. Фролов, Г.В.Фролов. Защищённый режим процессоров Intel 80286/ 80386/ 80486. Практическое руководство по использованию защищённого режима. Москва, Диалог-МИФИ, 2003.
6. Питер Абель. Язык Ассемблера для IBM PC и программирования. М., Высшая школа, 2003.

### **4.3. Общие требования к организации образовательного процесса**

Выполнение практических занятий предполагает деление группы по числу рабочих мест, оборудованных персональными компьютерами.

Обязательным условием допуска к производственной практике в рамках профессионального модуля является освоение учебной практики с целью получения первичных профессиональных навыков.

Консультации обучающихся проводятся согласно графика консультаций, составленному учебной частью.

Формой аттестации является дифференцированный зачет.

#### **4.4. Кадровое обеспечение образовательного процесса**

Требования к квалификации педагогических (инженерно-педагогических) кадров, обеспечивающих обучение по междисциплинарному курсу (курсам): наличие высшего профессионального образования, соответствующего профилю преподаваемого модуля.

Требования к квалификации педагогических кадров, осуществляющих руководство практикой:

Инженерно-педагогический состав: высшее инженерное образование, соответствующее профилю модуля.

Мастера: обязательная стажировка в профильных организациях не реже 1-го раза в 3 года. Опыт деятельности в организациях соответствующей профессиональной сферы является обязательным.

## 5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ (ВИДА ПРОФЕССИОНАЛЬНОЙ ДЕЯТЕЛЬНОСТИ)

**5.1. Контроль и оценка** результатов освоения МДК (практик) осуществляется преподавателем (руководителем практик) в процессе проведения практических занятий и лабораторных работ, контрольных работ, тестирования, выполнения обучающимися индивидуальных заданий, проектов, исследований и д.р..

Результаты (освоенные профессиональные компетенции)	Основные показатели оценки результата	Формы и методы контроля и оценки
ПК 1.1. Выполнять разработку спецификаций отдельных компонент.	<p><i>Точность определения основных этапов разработки программного обеспечения;</i></p> <p><i>Правильность применения основных принципов технологии структурного и объектно-ориентированного программирования;</i></p> <p><i>Правильность оформления документации на программные средства;</i></p> <p><i>Правильность и точность разработки алгоритма поставленной задачи</i></p>	<p><i>Текущий контроль в форме:</i></p> <ul style="list-style-type: none"> <li>- защиты лабораторных и практических занятий;</li> <li>- контрольных работ по темам МДК.</li> </ul> <p><i>Зачеты по производственной практике и по каждому из разделов профессионального модуля.</i></p> <p><i>Комплексный экзамен по профессиональному модулю.</i></p>
ПК 1.2. Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля.	<p><i>Правильность применение основных принципов технологии структурного и объектно-ориентированного программирования;</i></p> <p><i>Правильность и точность разработки кода программного модуля на современных языках программирования;</i></p> <p><i>Точность создания программы по разработанному алгоритму как отдельного модуля;</i></p> <p><i>Правильность разработки кода программного продукта на основе готовой спецификации на уровне модуля;</i></p>	<p><i>Текущий контроль в форме:</i></p> <ul style="list-style-type: none"> <li>- защиты лабораторных и практических занятий;</li> <li>- контрольных работ по темам МДК.</li> </ul> <p><i>Зачеты по производственной практике и по каждому из разделов профессионального модуля.</i></p> <p><i>Комплексный экзамен по профессиональному модулю.</i></p>
ПК 1.3. Выполнять отладку программных модулей с использованием специализированных	<p><i>Правильность применения основных принципов отладки и тестирования программных продуктов;</i></p>	<p><i>Текущий контроль в форме:</i></p> <ul style="list-style-type: none"> <li>- защиты лабораторных и практических занятий;</li> <li>- контрольных работ по темам МДК.</li> </ul>

программных средств.	<p><i>Точность использования инструментальных средств на этапе отладки программного продукта;</i></p> <p><i>Правильность отладки и тестирование программы на уровне модуля;</i></p>	<p><i>Зачеты по производственной практике и по каждому из разделов профессионального модуля.</i></p> <p><i>Комплексный экзамен по профессиональному модулю.</i></p>
ПК 1.4. Выполнять тестирование программных модулей.	<p><i>Проведения тестирования программного модуля по определенному сценарию;</i></p> <p><i>Правильность выполнения отладки и тестирование программы на уровне модуля;</i></p>	<p><i>Текущий контроль в форме:</i></p> <ul style="list-style-type: none"> <li><i>- защиты лабораторных и практических занятий;</i></li> <li><i>- контрольных работ по темам МДК.</i></li> </ul> <p><i>Зачеты по производственной практике и по каждому из разделов профессионального модуля.</i></p> <p><i>Комплексный экзамен по профессиональному модулю.</i></p>
ПК 1.5. Осуществлять оптимизацию программного кода модуля.	<p><i>Точность проведения оптимизации программного кода модуля по определенному сценарию;</i></p> <p><i>Правильность выполнения отладки и тестирование программы на уровне модуля;</i></p> <p><i>Правильность использования инструментальных средств на этапе отладки программного продукта</i></p>	<p><i>Текущий контроль в форме:</i></p> <ul style="list-style-type: none"> <li><i>- защиты лабораторных и практических занятий;</i></li> <li><i>- контрольных работ по темам МДК.</i></li> </ul> <p><i>Зачеты по производственной практике и по каждому из разделов профессионального модуля.</i></p> <p><i>Комплексный экзамен по профессиональному модулю.</i></p>
ПК 1.6. Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификаций.	<p><i>Правильность использования инструментальные средства для автоматизации оформления документации;</i></p> <p><i>Правильность определения и использование методов и средств разработки технической документации</i></p>	<p><i>Текущий контроль в форме:</i></p> <ul style="list-style-type: none"> <li><i>- защиты лабораторных и практических занятий;</i></li> <li><i>- контрольных работ по темам МДК.</i></li> </ul> <p><i>Зачеты по производственной практике и по каждому из разделов профессионального модуля.</i></p> <p><i>Комплексный экзамен по профессиональному модулю.</i></p>

*Результаты указываются в соответствии с паспортом программы и разделом 2. Перечень форм контроля должен быть конкретизирован с учетом специфики обучения по программе профессионального модуля.*

## **5.2 КОНТРОЛЬНЫЕ ВОПРОСЫ К ЗАЧЕТУ**

### **РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ**

#### **Раздел 1. Разработка спецификаций отдельных компонент**

1. История и назначение языка Си++.
2. Структура программы на языке С++. Примеры. Этапы создания исполняемой программы.
3. Состав языка С++. Константы и переменные С++.
4. Типы данных в С++.
5. Выражения. Знаки операций. Постфиксные и префиксные операции
6. Сводка операций: скобки, порядок вычислений, инкремент и декремент, преобразование типа.
7. Основные операторы С++ (присваивание, составные, выбора, циклов, перехода). Синтаксис, семантика, примеры.

#### **Раздел 2. Разработка кода программного продукта на основе готовых спецификаций на уровне модуля**

8. Массивы (определение, инициализация, способы перебора).
9. Сортировка массивов (простой обмен, простое включение, простой выбор).
10. Указатели. Операции с указателями. Примеры.
11. Одномерные массивы и указатели. Примеры.
12. Многомерные массивы и указатели. Примеры.
13. Символьная информация и строки. Функции для работы со строками (библиотечный файл string.h).
14. Функции в С++. Рекурсия. Примеры.
15. Прототип функции. Библиотечные файлы. Директива препроцессора #include.

#### **Раздел 3. 1. Отладка, тестирование и оптимизация программных модулей**

16. Передача одномерных массивов в функции. Примеры.
17. Передача многомерных массивов в функции. Примеры.
18. Передача строк в функции. Примеры.
19. Подставляемые функции. Примеры.
20. Ссылки на функции. Примеры.

## **5.3. КОНТРОЛЬНЫЕ ВОПРОСЫ К ЭКЗАМЕНУ**

### **РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ**

#### **Раздел 1. Разработка кода программного продукта на основе готовых спецификаций на уровне модуля**

1. Место языков ассемблера среди языков программирования.
2. Структура МП Intel 80x86: используемые регистры.
3. Структура МП Intel 80x86: операционное устройство и шинный интерфейс.
4. Размещение данных в памяти. Сегментация памяти.

5. Структура регистра флагов. Команды установки флагов.
6. Структура и форматы команд МП Intel 80x86. Команды пересылки данных.
7. Способы адресации в командах МП Intel 80x86.
8. Представление данных в IBM PC: целые числа.
9. Представление данных в IBM PC: двоично-десятичные числа.
10. Представление данных в IBM PC: алфавитно-цифровые данные.
11. Представление данных в IBM PC: вещественные данные.
12. Система команд МП: команды сложения и вычитания.
13. Команды умножения и деления чисел с ФТ.
14. Структура команд МП: базовая, индексная и косвенная адресации.
15. Логические команды обработки битов.
16. Команды сдвигов и их использование.
17. Команды передачи управления: безусловные переходы. Адресация в переходах.
18. Команды передачи управления: условные переходы.
19. Команды передачи управления: организация циклов.
20. Стек. Команды работы со стеком.
21. Элементарные конструкции языка ассемблера: алфавит, ключевые слова.
22. Элементарные конструкции языка ассемблера: числа, символьные данные.
23. Элементарные конструкции языка ассемблера: имена, метки.
24. Элементарные конструкции языка ассемблера: выражения и их использование.
25. Предложения языка ассемблера: комментарии.
26. Предложения языка ассемблера: команды.
27. Предложения языка ассемблера: директивы.
28. Структура файла ассемблер-программы. Директивы оформления программы.
29. Структура файла ассемблер-программы: односегментные и многосегментные файлы.
30. Использование прерываний в ассемблер-программах.
31. Операторы в командах языка ассемблера.
32. Блочная структура программы: правила описания и вызова процедур.
33. Блочная структура программы: расположение процедур в исходном файле.
34. Блочная структура программы: внутренние и внешние процедуры.
35. Способы передачи параметров между процедурой и вызывающей программой.

## **Раздел 2. Отладка, тестирование и оптимизация программных модулей**

36. Передача параметров между процедурой и вызывающей программой. Проблема сохранения регистров.
37. Программные пакеты MASM и TASM: этапы обработки задания (подготовка исходного файла и его трансляция).
38. Программные пакеты MASM и TASM: этапы обработки задания (компоновка объектного модуля и отладка программы).
39. Программные пакеты MASM и TASM: общие функции и различия.
40. Модели памяти и их использование в TASM.
41. Макросы: макроопределения и их использование.
42. Макросы: использование параметров и комментарии.

### **Перечень практических заданий к экзамену**

1. Задание: составить программу вычисления  $X$  в зависимости от значения символа  $ch$  по формуле, соответствующей варианту. Значение символа  $ch$  вводится с клавиатуры. Для контроля за правильностью работы программы использовать модуль IO.ASM. Формулы для расчета:

$$X = \begin{cases} (8g)/(5m), & \text{если } ch > 'c' \\ d+g-6, & \text{если } ch = 'c' \\ 3v-4d+100, & \text{если } ch < 'c' \end{cases}$$

2.Задание: составить программу вычисления X в зависимости от значения символа ch по формуле, соответствующей варианту. Значение символа ch вводится с клавиатуры. Для контроля за правильностью работы программы использовать модуль IO.ASM. Формулы для расчета:

$$X = \begin{cases} 3+vg-m, & \text{если } ch > 't' \\ (10d)/(7m), & \text{если } ch = 't' \\ m-7+3d, & \text{если } ch < 't' \end{cases}$$

3. Составить программу вычисления y по формуле:  $y=2m-7g+4-d*v$

4. Составить программу вычисления y по формуле:  $y=2m+8-g*d+5v$

5. Найти в DSEG: 1) 3-й по порядку нулевой байт; 2) 4-й по порядку код CR (0Dh); 3) 4-й байт из числа тех, которые ниже 20h; 4) 3-й по порядку код '\$'(24h);

6. Найти в DSEG: 1) байт, следующий за 3-м кодом ';' (3Bh); 2) 4-й байт из числа больших, чем 29h; 3) байт, следующий за 3-м отрицательным байтом;

7. Найти в DSEG: 1) байт, являющийся 4-м нечетным; 2) байт, следующий за 3-м кодом пробела (20h); 3) 3-й байт из числа тех, которые выше 10h;

8. Написать программу с использованием процедур, которая запрашивает строку (ввод с клавиатуры), и затем переводит все символы по следующему алгоритму: Если символ в нижнем регистре, перевести его в верхний регистр; если в верхнем – в нижний

9. Написать программу с использованием процедур, которая запрашивает строку (ввод с клавиатуры), и затем переводит все символы по следующему алгоритму: Вывести строку в обратном порядке

10. Написать программу с использованием процедур, которая запрашивает строку (ввод с клавиатуры), и затем переводит все символы по следующему алгоритму: Вывести строку, в закодированном виде, от каждого кода символа строки отнимается число 10.

11. Написать программу с использованием процедур, которая запрашивает строку (ввод с клавиатуры), и затем переводит все символы по следующему алгоритму: Удалить все символы в верхнем регистре;

12. Написать программу с использованием процедур, которая запрашивает строку (ввод с клавиатуры), и затем переводит все символы по следующему алгоритму: Найти позицию символа (вводится с клавиатуры) в строке и вывести позицию (и) в шестнадцатеричном виде.

13. Исходные данные:

- дата рождения студента в формате ДД-ММ-ГГ - числа – d, m, g (байт)
- возраст студента ( количество полных лет ) – число v (байт)

Задание: Составить программу вычисления y по формуле:  $y=5v-d*m+6g+3$

14. Исходные данные:

- дата рождения студента в формате ДД-ММ-ГГ - числа – d, m, g (байт)
- возраст студента ( количество полных лет ) – число v (байт)

Задание: Составить программу вычисления y по формуле:  $y=5d-g*v+7m-2$

15. Вычислить:  $(A*B-C)/D$

16. Найдите ошибку в следующих командах:

```
MOV DS,@DATA
MOV AX,ES
MOV DS,AX
MOV CS,AX
MOV (ячейка памяти), (ячейка памяти)
```

17. Найдите ошибку в следующих командах:

```
XCNG AX,BX
XCHG CX,BL
XCHG ES,DS
XCHG DATA1,AX
```

18. Вычисление результата выполнения арифметического выражения, в котором некоторые числа постоянны, а другие переменные.

Формула вычислений:  $X = (A * 2 + B * C) / (D - 3)$

19. Исходные данные:

- дата рождения студента в формате ДД-ММ-ГГ - числа – d, m, g (байт)
- возраст студента ( количество полных лет ) – число v (байт)

Задание: Составить программу вычисления у по формуле:  $y = 2v + 6d - m * g + 3$

20. Исходные данные:

- дата рождения студента в формате ДД-ММ-ГГ - числа – d, m, g (байт)
- возраст студента ( количество полных лет ) – число v (байт)

Задание: Составить программу вычисления у по формуле:  $y = g * m - 7d + v - 20$

21. Исходные данные:

- дата рождения студента в формате ДД-ММ-ГГ - числа – d, m, g (байт)
- возраст студента ( количество полных лет ) – число v (байт)

Задание: Составить программу вычисления у по формуле:  $y = g * m - 4d + 8v - 7$

22. Исходные данные:

- дата рождения студента в формате ДД-ММ-ГГ - числа – d, m, g (байт)
- возраст студента ( количество полных лет ) – число v (байт)

Задание: Составить программу вычисления у по формуле:  $y = 2 + m * d - 3g + 7v$