МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 44

УТВЕРЖДАЮ
Руководитель образовательной программы

старший преподаватель
(должность, уч. степень, звание)

Д.В. Куртяник
(инициалы, фамилия)

(подпись)
«16» февраля 2026 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Программирование на языках Ассемблера»
(Наименование дисциплины)

Код направления подготовки/ специальности	09.03.01
Наименование направления подготовки/ специальности	Информатика и вычислительная техника
Наименование направленности/ специализации	Компьютерные технологии, системы и сети
Форма обучения	заочная
Год приема	2026

Санкт-Петербург – 2026

Лист согласования рабочей программы дисциплины

Программу составил (а)

ст. преп. (должность, уч. степень, звание) | «16» февраля 2026 г (подпись, дата) | Д.В. Куртяник (инициалы, фамилия)

Программа одобрена на заседании кафедры № 44
«16» февраля 2026 г, протокол № 5-25/26

Заведующий кафедрой № 44

д.т.н.,проф. (уч. степень, звание) | «16» февраля 2026 г (подпись, дата) | М.Б. Сергеев (инициалы, фамилия)

Заместитель директора института №4 по методической работе

доц.,к.т.н. (должность, уч. степень, звание) | «16» февраля 2026 г (подпись, дата) | А.А. Фоменкова (инициалы, фамилия)

Аннотация

Дисциплина «Программирование на языках Ассемблера» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/ специальности 09.03.01 «Информатика и вычислительная техника» направленности/специализации «Компьютерные технологии, системы и сети». Дисциплина реализуется кафедрой «№44».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ПК-3 «Способен разрабатывать компоненты системных программных продуктов»

Содержание дисциплины охватывает круг вопросов, связанных с низкоуровневым программированием на ассемблере, архитектурой процессоров семейства Intel x86 (включая 16-битный и 32-битный режимы), принципами организации адресации оперативной памяти и регистрового устройства процессора. Изучаются команды, процедуры, макросы, технология Windows API, а также основы создания динамических библиотек и работа со структурами данных.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, лабораторные работы, самостоятельная работа обучающегося.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме зачета (4 семестр).

Общая трудоемкость освоения дисциплины составляет 3 зачетных единицы, 108 часов.

Язык обучения по дисциплине «русский »

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Получение обучающимися необходимых знаний, умений и навыков в области низкоуровневого программирования на языке ассемблера, формирование понимания архитектуры процессора и принципов работы вычислительных систем, как основы для развития профессиональных компетенций в сфере информационных технологий и смежных дисциплин.

1.2. Дисциплина входит в состав части, формируемой участниками образовательных отношений, образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

Категория (группа) компетенции	Код и наименование компетенции	Код и наименование индикатора достижения компетенции
Профессиональные компетенции	ПК-3 Способен разрабатывать компоненты системных программных продуктов	ПК-3.З.1 знать основы архитектуры, устройства и функционирования вычислительных систем ПК-3.У.1 уметь разрабатывать и осуществлять отладку программ, написанных на языке программирования низкого уровня ПК-3.В.1 владеть навыками изучения технической документации по языку программирования, системе команд процессора устройства, адресации памяти и регистров процессора устройства

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- «Информатика»,
- «Основы программирования».

Знания, полученные при изучении материала данной дисциплины, имеют как самостоятельное значение, так и используются при изучении других дисциплин:

- «Схемотехника»,
- «Организация ЭВМ и вычислительных систем»,
- «Микроконтроллерные системы»,
- «Теория автоматов».

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

Вид учебной работы	Всего	Трудоемкость по семестрам
		№4

1	2	3
Общая трудоемкость дисциплины, ЗЕ/ (час)	3/ 108	3/ 108
Из них часов практической подготовки	4	4
Аудиторные занятия, всего час.	8	8
в том числе:		
лекции (Л), (час)	4	4
практические/семинарские занятия (ПЗ), (час)		
лабораторные работы (ЛР), (час)	4	4
курсовой проект (работа) (КП, КР), (час)		
экзамен, (час)		
Самостоятельная работа, всего (час)	100	100
***Вид промежуточной аттестации*:** зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.)	Зачет,	Зачет,

4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий.

Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

Разделы, темы дисциплины	Лекции (час)	ПЗ (СЗ)	ЛР (час)	КП/КР (час)	СР (час)
Семестр 4					
Раздел 1. Архитектура процессоров Intel. Macro Assembler: IA-16 Тема 1.1. Язык ассемблера. Обзор ассемблеров. Тема 1.2. Режимы адресации. Тема 1.3. Регистры процессора. Тема 1.4. Оперативная память. Тема 1.5. Сборка ассемблерной программы. Тема 1.6. Основы MASM. Типы данных. Тема 1.7. Команды и директивы. Тема 1.8. Шаблон консольного приложения. знакомство с отладчиком. Тема 1.9. Команды пересылки данных и арифметические команды. Тема 1.10. Команды условного и безусловного перехода. Тема 1.11. Прямая и косвенная адресация. Тема 1.12. Циклические операции. Массивы. Сортировка массива. Тема 1.13. Стек. Тема 1.14. Прерывания. Тема 1.15. Процедуры. Тема 1.16. Макросы.	3		3		70

Раздел 2. Технология Windows API. Macro Assembler: IA-32 Тема 2.1. Архитектура IA-32. Переход на 32-битное программирование Тема 2.2. Технология Windows API Тема 2.3. Непосредственная работа с функциями Windows API Тема 2.4. Библиотека MASM32 Тема 2.5. Создание динамических библиотек Тема 2.6. Структуры и объединения	1		1		30
Итого в семестре:	4		4		100
Итого	4	0	4	0	100

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

Номер раздела	Название и содержание разделов и тем лекционных занятий
1	Раздел 1. Архитектура процессоров Intel. Macro Assembler: IA-16 Тема 1.1. Язык ассемблера. Обзор ассемблеров. Понятие ассемблера и его место среди языков программирования. Достоинства и недостатки ассемблерного кода. Обзор современных ассемблеров: TASM, MASM, FASM, NASM. Тема 1.2. Режимы адресации. Реальный и защищённый режимы работы процессора. Сегментная адресация памяти в реальном режиме: логический и физический адрес, сегментные регистры. Особенности защищённого режима и линейной модели памяти. Тема 1.3. Регистры процессора. Назначение и классификация 16-битных регистров процессора Intel 8086: регистры общего назначения, сегментные, индексные, указатели, регистр флагов. Расширение системы регистров в архитектуре IA-32. Тема 1.4. Оперативная память. Представление целых чисел со знаком и без знака в двоичной и шестнадцатеричной системах счисления; дополнительный код. Модель оперативной памяти как последовательности байт, адресация слов и двойных слов, порядок записи байтов (little-endian). Тема 1.5. Сборка ассемблерной программы. Понятие среды разработки, этапы компиляции (ассемблирование и

	компоновка). Использование командной строки и пакетных файлов для сборки программ MASM. Тема 1.6. Основы MASM. Типы данных. Синтаксис целочисленных и вещественных констант. Идентификаторы, команды и правило хорошего стиля. Обзор скалярных типов данных (BYTE, WORD, DWORD и др.), описание инициализированных и неинициализированных переменных. Тема 1.7. Команды и директивы. Различие между командами и директивами ассемблера. Директивы присваивания (= и EQU), счётчик команд $, оператор дублирования DUP. Понятие препроцессирования. Тема 1.8. Шаблон консольного приложения. Знакомство с отладчиком. Структура 16-битного консольного приложения: директивы .8086, .model, .stack, .data, .code. Основные окна и приёмы пошагового выполнения в отладчике Turbo Debugger. Тема 1.9. Команды пересылки данных и арифметические команды. Команды MOV, XCHG. Арифметические команды ADD, ADC, INC, SUB, SBB, DEC, MUL, IMUL, DIV, IDIV, NEG. Умножение и деление чисел разной размерности. Тема 1.10. Команды условного и безусловного перехода. Команды JMP, CMP и группа условных переходов для знаковых и беззнаковых чисел. Проверка отношения операндов по состоянию флагов. Тема 1.11. Прямая и косвенная адресация. Регистровая, непосредственная, прямая и косвенная адресация. Использование регистров BX, SI, DI, BP для формирования адреса. Индексная адресация и оператор PTR. Тема 1.12. Циклические операции. Массивы. Сортировка массива. Операторы LOOP, LOOPE, LOOPNE. Организация работы с одномерными массивами. Директивы TYPE, SIZEOF, LENGTHOF. Алгоритмы сортировки выбором. Тема 1.13. Стек. Организация стека в реальном режиме, регистры SS и SP. Команды PUSH, POP, PUSHA, POPA, PUSHF, POPF. Использование стека для временного хранения данных и передачи параметров. Тема 1.14. Прерывания. Классификация прерываний (аппаратные, внутренние, программные). Схема обработки прерывания. Основные функции прерывания int 21h для консольного ввода-вывода. Тема 1.15. Процедуры. Описание процедур с помощью PROC и ENDP. Передача параметров через регистры и

	стек. Соглашение stdcall. Директива INVOKE и прототипы. Локальные переменные процедуры. Тема 1.16. Макросы. Синтаксис макроопределений MACRO/ENDM, макросы с параметрами. Отличия от процедур. Подключение внешних файлов с макросами директивой INCLUDE.
2	Раздел 2. Технология Windows API. Macro Assembler: IA-32 Тема 2.1. Архитектура IA-32. Переход на 32-битное программирование. Основные отличия 16-битной и 32-битной архитектур. 32-битные регистры (EAX, EBX и др.), защищённый режим и линейная модель памяти. Расширение арифметических, циклических и стековых команд, особенности косвенной адресации в IA-32. Тема 2.2. Технология Windows API. Понятие интерфейса программирования приложений. Динамически подключаемые библиотеки (DLL), обзор ключевых системных библиотек (kernel32.dll, user32.dll). Шаблон 32-битного консольного приложения, назначение директив INCLUDE и INCLUDELIB. Знакомство с отладчиком OllyDbg. Тема 2.3. Непосредственная работа с функциями Windows API. Стандарты описания функций API, таблица соответствия типов C++ и MASM (BOOL→DWORD, HANDLE→DWORD и др.), директива typedef. Прототипы функций и их подключение. Работа с функциями консольного ввода-вывода: GetStdHandle, WriteConsole, ReadConsole, ExitProcess. Тема 2.4. Библиотека MASM32. Назначение и возможности библиотеки masm32.lib. Упрощённые процедуры консольного интерфейса: ClearScreen, StdOut, StdIn, locate. Особенности их вызова и передаваемые параметры. Тема 2.5. Создание динамических библиотек. Принципы технологии DLL, преимущества динамической компоновки. Разработка DLL без точки входа и с точкой входа (DllEntry). Файл определения модуля (.def), экспорт функций. Статическое и динамическое подключение библиотек: функции LoadLibrary, GetProcAddress, FreeLibrary. Тема 2.6. Структуры и объединения. Директивы STRUCT и UNION, описание полей, создание переменных структурного типа и инициализация значений. Доступ к полям через оператор «точка» и оператор PTR. Применение структур для передачи данных в процедуры и работа с массивами структур.

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

№ п/п	Темы практических занятий	Формы практических занятий	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Учебным планом не предусмотрено					
Всего					

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

№ п/п	Наименование лабораторных работ	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Семестр 4				
1	Изучение среды программирования MASM	1	1	1
2	Разработка программы в среде MASM	1	1	1
3	Разработка программы с использованием макрокоманд и процедур	1	1	1
4	Разработка встраиваемой программы	1	1	2
Всего		4		

4.5. Выполнение курсового проекта/ курсовой работы

Учебным планом не предусмотрено

4.6. Самостоятельная работа обучающихся

Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

Вид самостоятельной работы	Всего, час	Семестр 4, час
1	2	3
Изучение теоретического материала дисциплины (ТО)	40	40
Курсовое проектирование (КП, КР)		
Расчетно-графические задания (РГЗ)		
Выполнение реферата (Р)		
Подготовка к текущему контролю успеваемости (ТКУ)	40	40
Домашнее задание (ДЗ)		

Контрольные работы заочников (КРЗ)	10	10
Подготовка к промежуточной аттестации (ПА)	10	10
Всего:	100	100

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. разделов 6-11.

6. Перечень печатных и электронных учебных изданий

Перечень печатных и электронных учебных изданий приведен в таблице 8.

Таблица 8– Перечень печатных и электронных учебных изданий

Шифр/ URL адрес	Библиографическая ссылка	Количество экземпляров в библиотеке (кроме электронных экземпляров)
004.4 Р 83	Язык ассемблера : Уроки программирования [Текст] / П. И. Рудаков, К. Г. Финогенов. - М. : Диалог-МИФИ, 2001. - 640 с.	30
http://znanium.com/bookread2.php?book=409290 Режим доступа: для авторизованных пользователей.	Аблязов, Р. З. Программирование на ассемблере на платформе x86-64 [Электронный ресурс] / Р. З. Аблязов. - М.: ДМК Пресс, 2011. - 304 с.	

7. Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

URL адрес	Наименование
	Не предусмотрено

8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

№ п/п	Наименование
1	OllyDbg (Лицензия Shareware)
2	Microsoft Visual Studio (Лицензия freemium)

3	Электронная информационно-образовательная среда ГУАП «Интегрированная среда обучения» (https://pro.guap.ru/) разработана сотрудниками ГУАП (введена в эксплуатацию приказом ГУАП от 06.06.2017 № 05-215/17), перечень модулей и их функциональное назначение изложены по ссылке https://guap.ru/it/system/iso
4	Официальный сайт образовательной организации в сети «Интернет» (https://guap.ru/), разработан сотрудниками ГУАП (введён в эксплуатацию Приказом ГУАП от 23.03.2023 № 05-145/23)

8.2. Перечень информационно-справочных систем, используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

№ п/п	Наименование
1	Электронный каталог библиотеки ГУАП с доступом к базе полнотекстовых изданий (https://lib.guap.ru.), доступ через личный кабинет читателя библиотеки ГУАП
2	ЭБС Znanium (https://znanium.ru/), доступ через личный кабинет читателя библиотеки ГУАП, а также по IP-адресам ГУАП

9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице12.

Таблица 12 – Состав материально-технической базы

№ п/п	Наименование составной части материально-технической базы	Номер аудитории (при необходимости)
1	Мультимедийная лекционная аудитория	
2	Аудитория для проведения лабораторных работ	
3	Помещение для самостоятельной работы	

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средствдля проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

Вид промежуточной аттестации	Перечень оценочных средств
Зачет	Список вопросов; Тесты; Задачи.

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 –Критерии оценки уровня сформированности компетенций

Оценка компетенции 5-балльная шкала	Характеристика сформированных компетенций
«отлично» «зачтено»	Обучающийся: – глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно связывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий. – правильно выполнил от 90% до 100% тестовых заданий**.
«хорошо» «зачтено»	Обучающийся: – твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы; – не допускает существенных неточностей; – увязывает усвоенные знания с практической деятельностью направления; – аргументирует научные положения; – делает выводы и обобщения; – владеет системой специализированных понятий. – правильно выполнил от 70% до 89% тестовых заданий**.
«удовлетворительно» «зачтено»	– обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы; – допускает несущественные ошибки и неточности; – испытывает затруднения в практическом применении знаний направления; – слабо аргументирует научные положения; – затрудняется в формулировании выводов и обобщений; – частично владеет системой специализированных понятий. – правильно выполнил от 51% до 69% тестовых заданий**.
«неудовлетворительно» «не зачтено»	– обучающийся не усвоил значительной части программного материала; – допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении; – испытывает трудности в практическом применении знаний; – не может аргументировать научные положения; – не формулирует выводов и обобщений. – правильно выполнил менее 51% тестовых заданий**.

Примечание: **по решению кафедры процент правильно выполненных тестовых заданий может быть изменен.

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

№ п/п	Перечень вопросов (задач) для экзамена	Код индикатора
	Учебным планом не предусмотрено	

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.

Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

№ п/п	Перечень вопросов (задач) для зачета / дифф. зачета	Код индикатора
1.	Охарактеризуйте архитектуры IA 16 и IA 32: опишите особенности реального и защищённого режимов	ПК-3.З.1

	адресации, состав регистров, организацию памяти и принципы перехода от 16 битного к 32 битному программированию.	
2.	Опишите систему регистров процессора Intel 8086 и их расширение в IA 32, объясните назначение регистров общего назначения, сегментных, индексных и указателей.	ПК-3.З.1
3.	Перечислите основные директивы ассемблера, объясните их назначение и отличие от команд процессора.	ПК-3.З.1
4.	Опишите работу арифметических команд и их 32 битных расширений, объясните формирование флагов результата и особенности умножения/деления чисел разной размерности.	ПК-3.З.1
5.	Опишите способы определения одномерных массивов в MASM, объясните использование индексной адресации и директив TYPE, SIZEOF, LENGTHOF для получения характеристик массива.	ПК-3.З.1
6.	Сформулируйте понятие макроса, опишите синтаксис макроопределения, приведите пример создания макроса с параметрами и объясните ключевые отличия макросов от процедур.	ПК-3.З.1
7.	Опишите синтаксис описания процедур, объясните механизм вызова и назначение прототипов, перечислите способы передачи параметров.	ПК-3.З.1
8.	Перечислите и охарактеризуйте основные виды адресации операндов в ассемблере (регистровая, непосредственная, прямая, косвенная, индексная), приведите примеры каждой из них.	ПК-3.З.1
9.	Опишите команды безусловного и условных переходов, объясните их связь с флагами состояния и использование после команды сравнения CMP.	ПК-3.З.1
10.	Перечислите основные целочисленные и вещественные типы данных в, опишите их размерность и диапазоны допустимых значений для знаковых и беззнаковых форм.	ПК-3.З.1
11.	Опишите команды пересылки данных, укажите ограничения на операнды и приведите примеры обмена значениями между регистрами и памятью.	ПК-3.З.1
12.	Опишите организацию стека в реальном режиме IA 16, назначение регистров SS и SP, объясните работу команд PUSH и POP и приведите пример использования стека для временного хранения данных.	ПК-3.З.1
13.	Опишите реализацию циклических операций с помощью команды LOOP и её вариаций (LOOPE, LOOPNE), объясните роль регистра CX/ECX и правила организации цикла с заданным числом повторений.	ПК-3.З.1
14.	Объясните понятие прерывания, перечислите виды прерываний и опишите механизм вызова программного прерывания на примере функций int 21h для консольного ввода вывода.	ПК-3.З.1
15.	Дайте определение языка ассемблера, перечислите его основные достоинства и недостатки, а также назовите	ПК-3.З.1

	несколько распространённых ассемблеров и укажите их особенности.	
16.	Сформулируйте понятие Windows API, опишите роль динамически подключаемых библиотек (DLL) и перечислите основные системные библиотеки, используемые в консольных и оконных приложениях.	ПК-3.З.1
17.	Опишите три основных способа передачи аргументов в процедуры на ассемблере, сравните их и укажите особенности каждого способа.	ПК-3.З.1
18.	Опишите, каким образом ассемблерный код может быть интегрирован в программы на языках высокого уровня. Укажите основные правила и ограничения.	ПК-3.З.1
19.	Проанализируйте предложенный фрагмент кода, содержащий циклическую конструкцию (LOOP), найдите и исправьте ошибку, связанную с инициализацией счётчика или изменением регистра CX.	ПК-3.В.1
20.	Продемонстрируйте процесс ассемблирования и компоновки программы на MASM с помощью командной строки, укажите необходимые ключи и имена файлов для получения исполняемого модуля.	ПК-3.В.1
21.	Опишите и покажите на примере способ передачи параметров в процедуру через регистры: поясните, какие регистры используются, как передаются аргументы и возвращается результат.	ПК-3.В.1
22.	Объясните метод передачи параметров в процедуру через общую область памяти (глобальные переменные), опишите его особенности, достоинства и ограничения по сравнению с другими способами.	ПК-3.В.1
23.	Опишите механизм передачи параметров в процедуру через стек с использованием соглашения stdcall, приведите пример объявления процедуры с параметрами и её вызова (в том числе с директивой INVOKE).	ПК-3.В.1
24.	Используя отладчик, пошагово выполните предложенную программу, определите изменение значений регистров и памяти после заданных команд, найдите адреса переменных в дампе.	ПК-3.В.1
25.	Преобразуйте заданный фрагмент 16 битной программы в 32 битную с учётом особенностей архитектуры IA 32: замените регистры, измените модель памяти и способы вызова системных функций.	ПК-3.У.1
26.	Разработайте консольное приложение на MASM с использованием функций Windows API (GetStdHandle, WriteConsole, ExitProcess), которое выводит на экран заданную текстовую строку.	ПК-3.У.1
27.	Опишите порядок создания ассемблерной вставки в программу на языке C++, приведите пример вставки, выполняющей арифметическую операцию, и объясните правила доступа к переменным C++ из ассемблерного блока.	ПК-3.У.1
28.	Опишите процесс создания динамической библиотеки (DLL) на MASM: разработайте код библиотеки, составьте файл определения модуля (.def), выполните	ПК-3.У.1

	компиляцию и продемонстрируйте вызов экспортируемой функции из внешней программы.	
29.	Создайте отдельный файл с макроопределениями (.inc), включите в него несколько полезных макросов (например, для вывода строки и ожидания клавиши) и подключите его к основной программе директивой INCLUDE.	ПК-3.У.1
30.	Продемонстрируйте динамическое подключение системной DLL (kernel32.dll) с помощью функций LoadLibrary, GetProcAddress и FreeLibrary, вызовите функцию GetStdHandle для получения дескриптора стандартного вывода.	ПК-3.У.1
31.	Используя возможности библиотеки MASM32 (макросы или упрощённые процедуры), напишите программу, которая выводит форматированную строку с помощью макроса printf и ожидает нажатия любой клавиши.	ПК-3.У.1

Перечень тем для выполнения курсового проекта/ курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для выполнения курсового проекта / курсовой работы

№ п/п	Примерный перечень тем для выполнения курсового проекта/ курсовой работы
	Учебным планом не предусмотрено

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

№ п/п	Примерный перечень вопросов для тестов	Код индикатора
1.	Задание комбинированного типа с выбором одного верного ответа из четырёх предложенных и обоснованием выбора *Инструкция: Прочитайте текст и выберите один правильный ответ и запишите аргументы, обосновывающие выбор ответа* Выберите оператор языка ассемблера MASM, который используется для объявления сегмента данных: 1. code segment 2. data segment 3. stack segment 4. proc segment	ПК-3.З.1 ПК-3.У.1
2.	Задание комбинированного типа с выбором нескольких верных ответов из четырёх предложенных и обоснованием выбора *Инструкция: Прочитайте текст и выберите несколько правильных ответов и запишите аргументы, обосновывающие выбор ответа* Выберите все верные утверждения, которые относятся к директивам языка ассемблера MASM: 1. Директива end указывает на завершение программы. 2. Директива proc используется для объявления процедуры. 3. Директива data segment определяет начало сегмента данных.	ПК-3.З.1 ПК-3.В.1

<table>
<tr><td></td><td>4. Директива mov используется для объявления переменной.</td><td></td></tr>
<tr><td>3.</td><td>Задание комбинированного типа с выбором нескольких верных ответов из четырёх предложенных и обоснованием выбора
Инструкция: Прочитайте текст и выберите несколько правильных ответов и запишите аргументы, обосновывающие выбор ответа

Выберите все верные утверждения, которые описывают работу с регистрами в языке ассемблера MASM:
1. Регистр AX является 16-битным и часто используется для арифметических операций.
2. Регистр EAX является 32-битным и может использоваться для хранения адресов памяти.
3. Регистр CS отвечает за хранение данных в сегменте стека.
4. Регистр IP (Instruction Pointer) хранит адрес следующей выполняемой команды.</td><td>ПК-3.У.1
ПК-3.В.1</td></tr>
<tr><td>4.</td><td>Задание комбинированного типа с выбором одного верного ответа из четырёх предложенных и обоснованием выбора
Инструкция: Прочитайте текст и выберите один правильный ответ и запишите аргументы, обосновывающие выбор ответа

Укажите верное утверждение, описывающее назначение команды mov в языке ассемблера MASM:
1. Команда mov используется для выполнения арифметического сложения.
2. Команда mov копирует данные из одного места в другое (например, между регистрами, памятью и регистрами).
3. Команда mov завершает выполнение программы.
4. Команда mov используется для объявления сегмента данных.</td><td>ПК-3.З.1
ПК-3.В.1</td></tr>
<tr><td>5.</td><td>Задание на установление соответствия
Инструкция: Прочитайте текст и установите соответствие.

Сопоставьте элементы из первого столбца (команды и директивы MASM) и элементы из второго столбца (их назначение):

<table><tr><td>Элемент</td><td>Назначение</td></tr><tr><td>1. mov</td><td>A. Объявление сегмента данных.</td></tr><tr><td>2. data segment</td><td>B. Копирование данных между регистрами, памятью и регистрами.</td></tr><tr><td>3. end</td><td>C. Объявление процедуры.</td></tr><tr><td>4. proc</td><td>D. Завершение программы.</td></tr></table>Запишите выбранные буквы под соответствующими цифрами:

<table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td></td><td></td><td></td><td></td></tr></table></td><td>ПК-3.В.1</td></tr>
</table>

<table>
<tr><td>6.</td><td>Задание на установление соответствия
Инструкция: Прочитайте текст и установите соответствие.

Сопоставьте элементы из первого столбца (регистры в MASM) и элементы из второго столбца (их основное назначение):<table><tr><td>Регистр</td><td>Назначение</td></tr><tr><td>1. AX</td><td>A. Хранение адреса следующей команды для выполнения.</td></tr><tr><td>2. IP</td><td>B. Хранение данных и выполнение арифметических операций.</td></tr><tr><td>3. CS</td><td>C. Указание на сегмент кода, в котором находится выполняемая программа.</td></tr><tr><td>4. SP</td><td>D. Управление вершиной стека.</td></tr></table>Запишите выбранные буквы под соответствующими цифрами:<table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td></td><td></td><td></td><td></td></tr></table></td><td>ПК-3.У.1
ПК-3.В.1</td></tr>
<tr><td>7.</td><td>Задание на установление последовательности
Инструкция: Прочитайте текст и установите последовательность

Расположите в правильной последовательности шаги для вызова и возврата из процедуры на языке ассемблера MASM:
1. Использовать команду call
2. Передать аргументы в процедуру (например, через регистры или стек)
3. Выполнить код процедуры
4. Использовать команду ret

Запишите соответствующую последовательность цифр слева направо.</td><td>ПК-3.В.1</td></tr>
<tr><td>8.</td><td>Задание на установление последовательности
Инструкция: Прочитайте текст и установите последовательность

Расположите в правильной последовательности шаги, которые выполняет процессор при обработке команды mov ax, [bx] на языке ассемблера MASM в архитектуре x86:
1. Прочитать данные из памяти по адресу
2. Вычислить адрес памяти, используя значение регистра BX
3. Обновить регистр IP (Instruction Pointer) для перехода к следующей команде.
4. Загрузить прочитанные данные в регистр AX

Запишите соответствующую последовательность цифр слева направо</td><td>ПК-3.У.1</td></tr>
<tr><td>9.</td><td>Задание открытого типа с развёрнутым ответом
Инструкция: Прочитайте текст и запишите развёрнутый обоснованный ответ

Объясните, как работает механизм передачи аргументов в процедуру через стек в языке ассемблера MASM. Опишите последовательность действий, включая подготовку стека,</td><td>ПК-3.У.1</td></tr>
</table>

<table>
<tr><td></td><td>передачу аргументов, их использование внутри процедуры и восстановление стека после завершения процедуры. Приведите пример кода, иллюстрирующий этот процесс.</td><td></td></tr>
<tr><td>10.</td><td>Задание открытого типа с развёрнутым ответом
Инструкция: Прочитайте текст и запишите развёрнутый обоснованный ответ

Объясните, как реализовать цикл на языке ассемблера MASM для обработки массива данных. Опишите последовательность действий, включая инициализацию регистров, организацию цикла, обработку элементов массива и завершение цикла. Приведите пример кода, иллюстрирующий этот процесс.</td><td>ПК-3.У.1
ПК-3.В.1</td></tr>
<tr><td>11.</td><td>Задание комбинированного типа с выбором одного верного ответа из четырёх предложенных и обоснованием выбора
Инструкция: Прочитайте текст и выберите один правильный ответ и запишите аргументы, обосновывающие выбор ответа

Какая из перечисленных арифметических команд языка ассемблера MASM выполняет умножение двух целочисленных операндов с учётом знака?
1. add
2. imul
3. div
4. inc</td><td>ПК-3.З.1
ПК-3.У.1</td></tr>
<tr><td>12.</td><td>Задание комбинированного типа с выбором нескольких верных ответов из четырёх предложенных и обоснованием выбора
Инструкция: Прочитайте текст и выберите несколько правильных ответов и запишите аргументы, обосновывающие выбор ответа

Выберите все утверждения, которые верно описывают работу стека в реальном режиме IA-16:
1. Команда push уменьшает значение регистра SP на 2, затем записывает операнд в память.
2. Стек растёт в сторону старших адресов.
3. Команда pop считывает значение из памяти по адресу SS:SP, затем увеличивает SP на 2.
4. Ширина стека в реальном режиме составляет 4 байта.</td><td>ПК-3.З.1
ПК-3.В.1</td></tr>
<tr><td>13.</td><td>Задание на установление соответствия
Инструкция: Прочитайте текст и установите соответствие.

Сопоставьте макросы или упрощённые процедуры библиотеки MASM32 (левый столбец) с их назначением (правый столбец):
<table><tr><td>Макрос</td><td>Назначение</td></tr><tr><td>1. cls</td><td>A. Очистка экрана консоли</td></tr><tr><td>2. print</td><td>B. Вывод строки на экран</td></tr><tr><td>3. inkey</td><td>C. Форматированный вывод строк и чисел</td></tr><tr><td>4. printf</td><td>D. Ожидание нажатия любой клавиши</td></tr></table>Запишите выбранные буквы под соответствующими цифрами:
<table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td></td><td></td><td></td><td></td></tr></table></td><td>ПК-3.В.1</td></tr>
<tr><td>14.</td><td>Задание на установление последовательности</td><td>ПК-3.У.1</td></tr>
</table>

<table>
<tr><td></td><td>Инструкция: Прочитайте текст и установите последовательность

Расположите в правильной последовательности шаги, необходимые для вызова функции WriteConsole из Windows API в 32-битном приложении на MASM:
1. Вызвать функцию WriteConsole через invoke
2. Получить дескриптор стандартного вывода с помощью GetStdHandle
3. Загрузить адрес строки и количество символов в регистры/стек
4. Завершить процесс через ExitProcess

Запишите соответствующую последовательность цифр слева направо.</td><td></td></tr>
<tr><td>15.</td><td>Задание на установление соответствия
Инструкция: Прочитайте текст и установите соответствие.

Сопоставьте вид адресации (левый столбец) с соответствующим примером команды (правый столбец):
<table><tr><td>Вид адресации</td><td>Пример команды</td></tr><tr><td>1. Непосредственная</td><td>A. mov ax,[bx+4]</td></tr><tr><td>2. Прямая</td><td>B. mov ax,[bx]</td></tr><tr><td>3. Косвенная базовая</td><td>C. mov ax,10</td></tr><tr><td>4. Индексная со смещением</td><td>D. mov ax,[Var]</td></tr></table>Запишите выбранные буквы под соответствующими цифрами:
<table><tr><td>1</td><td>2</td><td>3</td><td>4</td></tr><tr><td></td><td></td><td></td><td></td></tr></table></td><td>ПК-3.В.1</td></tr>
<tr><td>16.</td><td>Задание комбинированного типа с выбором одного верного ответа из четырёх предложенных и обоснованием выбора
Инструкция: Прочитайте текст и выберите один правильный ответ и запишите аргументы, обосновывающие выбор ответа

Какое утверждение верно характеризует защищённый режим работы процессора IA-32?
1. Адресное пространство программы ограничено 1 Мбайт.
2. Используется сегментная адресация с 16-битными сегментными регистрами.
3. Программа может адресовать до 4 Гбайт линейного адресного пространства.
4. Запрещено использование 32-битных регистров общего назначения.</td><td>ПК-3.З.1
ПК-3.У.1</td></tr>
<tr><td>17.</td><td>Задание комбинированного типа с выбором нескольких верных ответов из четырёх предложенных и обоснованием выбора
Инструкция: Прочитайте текст и выберите несколько правильных ответов и запишите аргументы, обосновывающие выбор ответа

Выберите все верные утверждения, описывающие различия между макросами и процедурами в MASM:</td><td>ПК-3.З.1
ПК-3.В.1</td></tr>
</table>

	1. Тело макроса подставляется в код каждый раз при вызове, а тело процедуры существует в единственном экземпляре. 2. Макросы, как и процедуры, автоматически сохраняют значения регистров в стеке. 3. Процедура вызывается командой call, а макрос – указанием его имени. 4. Макросы могут принимать параметры, а процедуры – нет.	
18.	Задание открытого типа с развёрнутым ответом *Инструкция: Прочитайте текст и запишите развёрнутый обоснованный ответ* Опишите назначение и основные этапы работы программного прерывания int 21h в DOS-приложениях. Для функции вывода строки 09h укажите: какой регистр должен содержать номер функции, в каком регистре передаётся адрес строки и каким символом должна заканчиваться выводимая строка.	ПК-3.У.1
19.	Задание открытого типа с развёрнутым ответом *Инструкция: Прочитайте текст и запишите развёрнутый обоснованный ответ* Объясните, как реализовать вызов ассемблерной вставки в программе на языке C++. Укажите, какой синтаксис используется для вставки кода, какие ограничения существуют при доступе к переменным C++ и как возвратить результат из вставки. Приведите простой пример вставки, выполняющей сложение двух целочисленных переменных.	ПК-3.В.1
20.	Задание на установление последовательности *Инструкция: Прочитайте текст и установите последовательность* Расположите в правильной последовательности шаги по созданию и использованию динамической библиотеки (DLL) в MASM: 1. Написать код библиотеки (процедуры) и файл определения модуля (.def) 2. Скомпилировать библиотеку с ключом /DLL 3. Подключить библиотеку импорта (.lib) или использовать динамическую загрузку (LoadLibrary) 4. Вызвать экспортируемую функцию из основной программы Запишите соответствующую последовательность цифр слева направо.	ПК-3.В.1
21.	Задание комбинированного типа с выбором одного верного ответа из четырёх предложенных и обоснованием выбора *Инструкция: Прочитайте текст и выберите один правильный ответ и запишите аргументы, обосновывающие выбор ответа* Какой тип данных в MASM соответствует типу HANDLE из Windows API? 1. BYTE 2. WORD 3. DWORD	ПК-3.З.1

	4. REAL4	

Примечание: система оценивания тестовых заданий.

Оценка тестовых заданий балльная шкала	Характеристика заданий
Полное совпадение с верным ответом оценивается 1 баллом / неверный ответ или его отсутствие – 0 баллов.	1 тип) Задание комбинированного типа с выбором одного верного ответа из четырёх предложенных и обоснованием выбора считается верным, если правильно указана цифра и приведены конкретные аргументы, используемые при выборе ответа. Задание с выбором одного верного ответа из четырёх предложенных считается верным, если правильно указана цифра
Полное совпадение с верным ответом оценивается 1 баллом, если допущены ошибки или ответ отсутствует 0 баллов.	2 тип) Задание комбинированного типа с выбором нескольких вариантов ответа из предложенных и развёрнутым обоснованием выбора считается верным, если правильно указаны цифры и приведены конкретные аргументы, используемые при выборе ответов. Задание с выбором нескольких верных ответов из четырёх предложенных считается верным, если правильно указаны цифры
«Полное совпадение с верным ответом оценивается 1баллом, неверный ответ или его отсутствие - 0 баллов»	3 тип) Задание закрытого типа на установление соответствия считается верным, если установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого столбца)
«Полное совпадение с верным ответом оценивается 1баллом, если допущены ошибки или ответ отсутствует – 0 баллов.»	4 тип) Задание закрытого типа на установление последовательности считается верным, если правильно указана вся последовательность цифр.
«Правильный ответ за задание оценивается в 3 балла, если допущена одна ошибка \ неточность \ ответ правильный, но не полный - 1 балл, если допущено более 1 ошибки \ ответ неправильный \ ответ отсутствует – 0 баллов».	5 тип) Задание открытого типа с развёрнутым ответом считается верным, если ответ совпадает с эталонным по содержанию и полноте.

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

№ п/п	Перечень контрольных работ
1	Архитектура процессоров Intel.
2	Технология Windows API.

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

11. Методические указания для обучающихся по освоению дисциплины

11.1. Методические указания для обучающихся по освоению лекционного материала.

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимися лекционного материала:

– получение современных, целостных, взаимосвязанных знаний, уровень которых определяется целевой установкой к каждой конкретной теме;

– получение опыта творческой работы совместно с преподавателем;

– развитие профессионально-деловых качеств, любви к предмету и самостоятельного творческого мышления.

– появление необходимого интереса, необходимого для самостоятельной работы;

– получение знаний о современном уровне развития науки и техники и о прогнозе их развития на ближайшие годы;

– научиться методически обрабатывать материал (выделять главные мысли и положения, приходить к конкретным выводам, повторять их в различных формулировках);

– получение точного понимания всех необходимых терминов и понятий.

Лекционный материал может сопровождаться демонстрацией слайдов и использованием раздаточного материала при проведении коротких дискуссий об особенностях применения отдельных тематик по дисциплине.

Структура предоставления лекционного материала:

– изложение теоретических вопросов, связанных с рассматриваемой темой;

– демонстрация примеров использования рассмотренных на лекции команд, операторов и директив Ассемблера;

– обобщение изложенного материала;

– ответы на возникающие вопросы по теме лекции.

11.2. Методические указания для обучающихся по участию в семинарах

Учебным планом не предусмотрено.

11.3. Методические указания для обучающихся по прохождению практических занятий

Учебным планом не предусмотрено.

11.4. Методические указания для обучающихся по выполнению лабораторных работ

В ходе выполнения лабораторных работ обучающийся должен углубить и закрепить знания, практические навыки, овладеть современной методикой и техникой эксперимента в соответствии с квалификационной характеристикой обучающегося. Выполнение лабораторных работ состоит из экспериментально-практической, расчетно-аналитической частей и контрольных мероприятий.

Выполнение лабораторных работ обучающимся является неотъемлемой частью изучения дисциплины, определяемой учебным планом, и относится к средствам, обеспечивающим решение следующих основных задач обучающегося:

– приобретение навыков исследования процессов, явлений и объектов, изучаемых в рамках данной дисциплины;

– закрепление, развитие и детализация теоретических знаний, полученных на лекциях;

– получение новой информации по изучаемой дисциплине;

– приобретение навыков самостоятельной работы с лабораторным оборудованием и приборами.

Задание и требования к проведению лабораторных работ

Вариант задания по каждой лабораторной работе обучающийся получает в соответствии с номером в списке группы. Перед проведением лабораторной работы обучающемуся следует внимательно ознакомиться с методическими указаниями по ее выполнению. В соответствии с заданием обучающийся должен выполнить указанную последовательность действий, получить требуемые результаты, оформить и защитить отчёт по лабораторной работе.

Структура и форма отчета о лабораторной работе

Отчёт о лабораторной работе должен включать в себя: титульный лист, формулировку задания, теоретические положения, используемые при выполнении лабораторной работы, описание процесса выполнения лабораторной работы, полученные результаты и выводы

Требования к оформлению отчета о лабораторной работе

По каждой лабораторной работе выполняется отдельный отчёт. Титульный лист оформляется в соответствии с шаблоном (образцом), приведённым на сайте ГУАП (www.guap.ru) в разделе «Нормативная документация». Текстовые и графические материалы оформляются в соответствии с действующими ГОСТами и требованиями, приведёнными на сайте ГУАП (www.guap.ru) в разделе «Нормативная документация».

11.5. Методические указания для обучающихся по выполнению курсового проекта/ курсовой работы

Учебным планом не предусмотрено.

11.6. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

Для обучающихся по заочной форме обучения выполнение контрольных работ является элементом текущего контроля успеваемости и самостоятельной работы.

В процессе выполнения самостоятельной работы у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет ему развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическими материалами, направляющими самостоятельную работу обучающихсяявляются:

– учебно-методический материал по дисциплине;

– методические указания по выполнению контрольных работ (для обучающихся по заочной форме обучения).

11.7. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

Методом проведения текущего контроля является сдача и защита четырёх лабораторных работ с соблюдением графика, установленного в начале семестра. График формируется через функционал личного кабинета «Запретить загружать отчёт после предельной даты».

Лабораторные работы оцениваются по бинарной шкале (Принято/Не принято; Зачтено/Не зачтено). За принятую работу начисляется один балл.

При нарушении сроков отчётности обучающийся получает ноль баллов за сданные работы.

11.8. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

– зачет – это форма оценки знаний, полученных обучающимся в ходе изучения учебной дисциплины в целом или промежуточная (по окончании семестра) оценка знаний обучающимся по отдельным разделам дисциплины с аттестационной оценкой «зачтено» или «не зачтено».

ПА проводится устно. Обучающийся получает вопрос из списка вопросов (задач).

Результаты ТКУ могут быть учтены при оценивании результатов освоения дисциплины, вплоть до выставления аттестационной оценкой «зачтено» без получения вопроса во время ПА. Для этого необходимо сдать все лабораторные работы и набрать не менее трёх баллов по результатам ТКУ.

За каждую несданную лабораторную работу, обучающийся получает дополнительный вопрос по соответствующей теме во время сдачи ПА.

Лист внесения изменений в рабочую программу дисциплины

Дата внесения изменений и дополнений. Подпись внесшего изменения	Содержание изменений и дополнений	Дата и № протокола заседания кафедры	Подпись зав. кафедрой