

**Федеральное государственное автономное  
образовательное учреждение высшего образования  
«Национальный исследовательский Нижегородский государственный университет  
им. Н.И. Лобачевского»**

**Институт информационных технологий, математики и механики**  
(факультет / институт / филиал)

УТВЕРЖДЕНО  
решением Ученого совета ННГУ  
протокол от  
30.11.2022 №13

**Рабочая программа дисциплины**

**Низкоуровневое программирование**

*(наименование дисциплины (модуля))*

Уровень высшего образования

бакалавриат

*(бакалавриат / магистратура / специалитет)*

Направление подготовки / специальность

**09.03.03 Прикладная информатика**

*(указывается код и наименование направления подготовки / специальности)*

Направленность образовательной программы

**Прикладная информатика в области принятия решений**

*(указывается профиль / магистерская программа / специализация)*

Форма обучения

очная

*(очная / очно-заочная / заочная)*

Нижний Новгород

2022

## 1. Место дисциплины в структуре ОПОП

Дисциплина относится к части, формируемой участниками образовательных отношений

№ варианта	Место дисциплины в учебном плане образовательной программы	Стандартный текст для автоматического заполнения в конструкторе РПД
2	Блок 1. Дисциплины (модули) Часть, формируемая участниками образовательных отношений	Дисциплина Б1.В.05 Низкоуровневое программирование. относится к части ООП направления подготовки 09.03.03 Прикладная информатика, формируемой участниками образовательных отношений.

## 2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции* (код, содержание индикатора)	Результаты обучения по дисциплине**	
<b>ПК-4.</b> Способен проводить исследование и описание процессов принятия решений в конкретной предметной (проблемной) области с применением современных информационных технологий, в том числе основанных на моделях и методах искусственного интеллекта	<b>ПК-4.1.</b> Демонстрирует знание современных моделей и методов интеллектуальной поддержки процессов принятия решений.	Знать этапы жизненного цикла программного обеспечения и методологии разработки программного обеспечения на языке ассемблера.	собеседование
	<b>ПК-4.2.</b> Демонстрирует умение применять системный подход к исследованию и описанию предметной (проблемной) области, формированию требований к ИС (ИИС) с учетом возможностей интеллектуальных технологий.	Уметь применять методологию разработки программного обеспечения на языке ассемблера без использования коммерческих IDE.	собеседование
	<b>ПК-4.3.</b> Имеет практический опыт исследования и описания конкретной предметной	Владеть навыками отладки программ в отладчиках на языке ассемблера.	задачи

	области, разработки технического задания, эскизного и технического проектов ИС (ИИС).		
<b>ПК-5.</b> Способен проектировать интеллектуальные ИС (ИИС) по видам обеспечения	<b>ПК-5.1.</b> Демонстрирует знание современных технологий проектирования ИИС.	Знать основы функционирования ОС Windows, структуру и принципы функционирования пользовательского приложения для ОС Windows.	собеседование
	<b>ПК-5.2.</b> Демонстрирует умение проектировать архитектуру ИИС по видам обеспечения.	Уметь применять теоретические знания для разработки программ с оконным и консольным интерфейсом на языке ассемблера для ОС Windows.	собеседование
	<b>ПК-5.3.</b> Имеет практический опыт проектирования конкретной ИИС по видам обеспечения.	Владеть основами языка ассемблера для решения системных и прикладных задач	задачи

### 3 Структура и содержание дисциплины

#### 3.1 Трудоемкость дисциплины

	<b>очная форма обучения</b>
<b>Общая трудоемкость</b>	<b>4 ЗЕТ</b>
<b>Часов по учебному плану</b>	<b>144</b>
<b>в том числе</b>	
<b>контактная работа:</b>	<b>91</b>
- занятия лекционного типа	<b>18</b>
- занятия лабораторные работы	<b>54</b>
- занятия семинарского типа	<b>18</b>
- текущий контроль (КСР)	<b>1</b>
<b>самостоятельная работа</b>	<b>53</b>
<b>Промежуточная аттестация – зачет</b>	

### 3.2.Содержание дисциплины

Наименование и краткое содержание разделов и тем дисциплины,  форма промежуточной аттестации по дисциплине	Всего (часы)	В том числе				Самостоятельная работа студента часы
		контактная работа (работа во взаимодействии с преподавателем), часы,из них				
		Занятия лекционного типа	Занятия практического типа	Занятия лабораторного типа	Всего контактных часов	Всего СРС
Последовательность выполнения команд для модели фон-Неймановской архитектуры компьютера, фундаментальные механизмы управления памятью ОС Windows	15	5			5	10
Методологии разработки и отладки низкоуровневого программного обеспечения	2	2			2	
Основы языка ассемблера, формы представления данных	40	6		34	40	
Программирование консольных приложений, работа с файлами, графические приложения Windows	13	3		10	13	
Вопросы системного программирования	44	1			1	43
Основы анализа и отладки кода программ	29	1	18	10	29	
В т.ч. текущий контроль	1				1	
Промежуточная аттестация - зачет						

Текущий контроль успеваемости реализуется в рамках лабораторных работ.

Промежуточная аттестация проходит в традиционной форме (зачет).

#### 4 Учебно-методическое обеспечение самостоятельной работы обучающихся

Для самоконтроля у студента имеется возможность изучения материала в дистанционном управляемом курсе (требуется авторизация): <https://e-learning.unn.ru/course/view.php?id=7041>

Самостоятельная работа студента при изучении дисциплины «Низкоуровневое программирование» включает изучение тем, решение домашних заданий и подготовку к зачету.

1. Тематика самостоятельной работы
2. Вопросы системного программирования.
3. Устройство операционной системы Windows.
4. Win32 API.
5. Вычисление в двоичной и шестнадцатеричной системах счисления.

#### Вопросы для самостоятельной работы

1. Роль процессора в компьютерной системе. Взаимосвязь операционной системы и аппаратной части компьютера.
2. Регистры процессора и их структура. Флаги состояний процессора.
3. Структура и доступ к оперативной памяти компьютера.
4. Взаимосвязь команд языка ассемблера и машинного кода.

5. Вычисление в двоичной и шестнадцатеричной системах счислений.
6. Типы утверждений в языке ассемблера.
7. Основные этапы получения различных типов выполняемых программ.
8. Интерпретация и инициализация переменных.
9. Стек, указатели и косвенная адресация.
10. Прерывания. Подпрограммы сервисных прерываний. Программные и аппаратные прерывания.
11. Логические операторы языка ассемблера.
12. Преимущества макроопределений перед процедурами с точки зрения их вызовов.
13. Структуры и функции Win32API для написания графических приложений для ОС Windows.
14. Типы многозадачности. Создание и управление потоком. Взаимодействие потоков.
15. Средства, упрощающие файловую обработку.
16. Способы взаимодействия и обмен информацией между запущенными приложениями.
17. Средства, упрощающие операции над группой файлов и каталогами.
18. Адресное пространство процесса. Функции управления памятью.
19. Методика отладки программ.
20. Восстановление исходного текста программы по ассемблерному коду.
21. Способы и приемы исправления исполняемых модулей.

**5 Фонд оценочных средств для промежуточной аттестации по дисциплине, включающий:**

**Описание шкал оценивания результатов обучения по дисциплине**

Уровень сформированности компетенций (индикатор а достижения компетенции)	Шкала оценивания сформированности компетенций						
	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	Не зачтено		зачтено				
<u>Знания</u>	Отсутствие знаний теоретического материала.  Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущено много негрубых ошибок.	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить	При решении стандартных задач не продемонстрированы	Продemonstrированы основные умения. Решены типовые	Продemonstrированы все основные умения. Решены все основные	Продemonstrированы все основные умения. Решены все основные	Продemonstrированы все основные умения, решены все основные	Продemonstrированы все основные умения,. Решены все основные

	наличие умений вследствие отказа обучающегося от ответа	основные умения.  Имели место грубые ошибки.	задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме.	задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами.	задачи . Выполнены все задания, в полном объеме, но некоторые с недочетами.	задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме.	задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие владения материалом. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки.  Имели место грубые ошибки.	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продemonstr ированы базовые навыки  при решении стандартных задач с некоторыми недочетами	Продemonstr ированы базовые навыки  при решении стандартных задач без ошибок и недочетов.	Продemonstr ированы навыки  при решении нестандартных задач без ошибок и недочетов.	Продemonstr ирован творческий подход к решению нестандартных задач

### Шкала оценки при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	Превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно»
	Отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна компетенция сформирована на уровне «отлично»
	Очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо», при этом хотя бы одна компетенция сформирована на уровне «очень хорошо»
	Хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «хорошо»
	Удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
	Неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно», ни одна из компетенций не сформирована на уровне «плохо»

не зачтено		
	Плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

## 5.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения

### 5.2.1. Контрольные вопросы

Вопросы	Код формируемой компетенции
1. Роль процессора в компьютерной системе. Взаимосвязь операционной системы и аппаратной части компьютера.	ПК-4
2. Регистры процессора и их структура. Флаги состояний процессора.	ПК-4
3. Структура и доступ к оперативной памяти компьютера.	ПК-4
4. Взаимосвязь команд языка ассемблера и машинного кода.	ПК-4
5. Вычисление в двоичной и шестнадцатеричной системах счислений.	ПК-4
6. Основные этапы разработки на языке ассемблера различных программ с различными типами пользовательских интерфейсов.	ПК-4
7. Типы утверждений в языке ассемблера.	ПК-5
8. Стек, указатели и косвенная адресация.	ПК-5
9. Логические операторы языка ассемблера.	ПК-5
10. Методика отладки программ.	ПК-5
11. Адресное пространство процесса. Функции управления памятью.	ПК-5

### 5.2.2. Типовые тестовые задания для оценки сформированности компетенции

#### Примеры заданий для оценивания компетенций ПК-4 в виде лабораторных работ.

**Задача 1.** Выполнить пошаговую отладку процедуры, осуществляющую сложение двух чисел, предварительно помещенных в стек, с помощью косвенной адресации.

**Задача 2.** Выполнить пошаговую отладку процедуры, осуществляющую сложение двух чисел, принимаемых в качестве параметров процедуры. Использовать директиву `invoke` для вызова процедуры.

**Задача 3.** Выполнить пошаговую отладку процедуры, которая выводит строку символов на консоль, и принимает в качестве параметров строку и ее размер. Использовать прием резервирования места в стеке для локальных параметров процедуры. Использовать относительную адресацию для инициализации параметров API ф-ии вывода на консоль.

#### Примеры заданий для оценивания компетенций ПК-5 в виде лабораторных работ.

**Задача 1.** Создать процедуру, осуществляющую сложение двух чисел, предварительно помещенных в стек, с помощью косвенной адресации.

**Задача 2.** Создать процедуру, осуществляющую сложение двух чисел, принимаемых в качестве параметров процедуры. Использовать директиву `invoke` для вызова процедуры.

**Задача 3.** Создать процедуру, которая выводит строку символов на консоль, и принимает в качестве параметров строку и ее размер. Использовать прием резервирования места в стеке для локальных параметров процедуры. Использовать относительную адресацию для инициализации параметров API ф-ии вывода на консоль.

**Задача 4.** Создать программу поиска всех простых чисел в заданном диапазоне и вывести их на экран консольного приложения.

**Задача 5.** Создать программу умножения двух девятиразрядных десятичных чисел и вывести результат на экран консольного приложения.

**Задача 6.** Создать программу чтения ранее созданного файла, указав его имя в командной строке, и вывести содержимое файла на экран. В случае отсутствия файла или имени файла в командной строке - вывести предупреждение на экран консольного приложения.

**Задача 7.** Создать программу, использующую оконный интерфейс, обрабатывающую действия пользователя в интерфейсе и выводящую сообщение в диалоговое окно.

## **6. Учебно-методическое и информационное обеспечение дисциплины**

а) основная литература:

1. **Вострикова З. П.** - Программирование на языке ассемблера ЕС ЭВМ: [учеб. пособие для вузов]. - М.: Наука, 1985. - 304 с. (26 экз.)

## **7. Материально-техническое обеспечение дисциплины (модуля)**

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой, оснащенные оборудованием и техническими средствами обучения.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ **09.03.03 «Прикладная информатика».**

Автор \_\_\_\_\_ доцент Белокрылов П.Ю.

Рецензент \_\_\_\_\_ профессор Федосенко Ю.С.

Заведующий кафедрой \_\_\_\_\_ М.Х.Прилуцкий

Программа одобрена на заседании методической комиссии института информационных технологий, математики и механики

07.12.2022 протокол №4