

Белорусский государственный университет

**УТВЕРЖДАЮ**

Проректор по учебной работе  
и образовательным инновациям

  
О.Г. Прохоренко

«05» июля 2023 г.

Регистрационный № УД – 1111/б.

## **МАШИННО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ**

**Учебная программа учреждения высшего образования  
по учебной дисциплине для специальности:**

**6-05-0533-09 Прикладная математика**

2023 г.

Учебная программа составлена на основе ОСВО 6-05-0533-09-2023 специальности 6-05-0533-09 «Прикладная математика», примерного учебного плана, регистрационный № 6-05-05-020/пр. от 20.12.2022, учебных планов БГУ: № 6-5.3-57/01 от 15.05 2023 г., № 6-5.3-57/02 от 15.05 2023 г., № 6-5.3-57/03 от 15.05 2023 г., № 6-5.3-57/04 от 15.05 2023 г.

#### **СОСТАВИТЕЛИ:**

**Зенько Т.А.**, старший преподаватель кафедры технологий программирования факультета прикладной математики и информатики Белорусского государственного университета;

**Карпович Н.А.**, старший преподаватель кафедры технологий программирования факультета прикладной математики и информатики Белорусского государственного университета;

**Пазюра Е.В.**, старший преподаватель кафедры технологий программирования факультета прикладной математики и информатики Белорусского государственного университета.

#### **РЕЦЕНЗЕНТЫ:**

**В.М.Котов** – заведующий кафедрой дискретной математики и алгоритмики Белорусского государственного университета, доктор физико-математических наук, профессор.

**П.В. Гляков** – профессор кафедры информационных технологий в культуре Белорусского государственного университета культуры и искусства, кандидат физико-математических наук, профессор;

#### **РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:**

Кафедрой технологий программирования  
(протокол № 16 от 18 мая 2023 г.);

Научно-методическим советом БГУ  
(протокол № 9 от 29 июня 2023 г.)

Заведующий кафедрой



А.Н. Курбацкий



## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

### Цели и задачи учебной дисциплины

**Цели учебной дисциплины** «Машинно-ориентированное программирование»: сформировать знание основ машинно-ориентированного программирования на различных платформах, сформировать понимание того, как язык ассемблера связан с архитектурой компьютера, как организовать работу с регистрами процессора, сформировать умение реализации алгоритма на языке низкого уровня в той части задач, где существенно проявляется зависимость параметров качества программы от учета особенностей архитектуры компьютера.

Содержание дисциплины включает круг вопросов, связанных с разработкой программного обеспечения, обеспечивающего взаимодействие модулей, написанных на разных языках программирования. В качестве среды разработки могут быть использованы различные платформы с организацией взаимодействия языков низкого уровня типа ассемблера и языка программирования C++.

### Задачи учебной дисциплины

Основной задачей, решаемой при изучении учебной дисциплины «Машинно-ориентированное программирование», является подготовка специалиста, умеющего писать на языке программирования низкого уровня компактные, эффективные программные модули, взаимодействующие с аппаратным обеспечением компьютера, на основе знаний об особенностях структурно-функциональной организации компьютера.

### Место учебной дисциплины

В системе подготовки специалиста с общим высшим образованием для специальности 6-05-0533-09 «Прикладная математика» учебная дисциплина «Машинно-ориентированное программирование» относится **к модулю** «Программирование» государственного компонента.

Учебная программа составлена с учетом межпредметных **связей** и программ по дисциплинам: основой для изучения учебной дисциплины является дисциплина государственного компонента «Основы и методологии программирования» модуля «Программирование», дисциплина компонента учреждения образования «Дискретная математика и математическая логика» модуля «Дискретная математика и алгоритмика». Знания, полученные в учебной дисциплине, используются при изучении дисциплин государственного компонента «Разработка кроссплатформенных приложений», «Промышленное программирование», «Технология программирования» модуля «Программирование».

### Требования к компетенциям

Освоение учебной дисциплины «Машинно-ориентированное программирование» должно обеспечить формирование следующих компетенций:

*универсальной* компетенции

УК-2. Решать стандартные задачи профессиональной деятельности на основе применения информационно-коммуникационных технологий;

**базовой профессиональной компетенции**

БПК-4. Применять навыки построения, анализа и тестирования алгоритмов и программ для решения типовых задач прикладной математики;

БПК-5. Применять при проектировании приложений такие парадигмы программирования как структурное, объектно-ориентированное и функциональное программирование, а также иные парадигмы, разрабатывать программное обеспечение в интегрированных средах разработки.

В результате освоения учебной дисциплины студент должен:

**знать:**

- механизмы ассемблирования и организацию ассемблерных программ;
- методы представления данных в вычислительных машинах и преобразования между внешним и внутренним представлением данных;
- методы эффективного использования знания архитектуры компьютера для оптимизации программ;
- методы тестирования и отладки программ;

**уметь:**

- программировать задачи средней сложности на языке ассемблера;
- оценивать эффективность различных машинно-ориентированных программно-технических решений;
- выбирать программные и технические средства реализации алгоритмов;
- разрабатывать программные приложения с заданной функциональностью и операционным окружением;

**владеть:**

- навыками программирования обработки числовой, логической и текстовой информации;
- навыками управления вычислительными процессами при программировании на ассемблере;
- навыками тестирования и отладки программ;
- навыками организации взаимодействия программных модулей, разрабатываемых на ассемблере, с модулями, разрабатываемыми на языках высокого уровня.

### **Структура учебной дисциплины**

Дисциплина изучается во 2-м семестре. Всего на изучение учебной дисциплины «Машинно-ориентированное программирование» отведено

— для очной формы получения высшего образования – 90 часов, в том числе 34 аудиторных часа, из них: лекции – 16 часов, лабораторные занятия – 16 часов, управляемая самостоятельная работа – 2 часа.

Трудоемкость учебной дисциплины составляет 3 зачётных единицы.

Форма промежуточной аттестации – зачет.

## СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

### Раздел 1. Принципы функционирования микропроцессоров

#### **Тема 1.1. *Архитектура компьютера***

Архитектура компьютера. Принстонская и гарвардская архитектура. Регистры процессора. Кэш-память процессора. Принципиальная схема компьютера на базе процессора x86/x64. Механизм исполнения машинных команд.

#### **Тема 1.2. *Память компьютера и представление данных в памяти.***

Представление целочисленных данных и адресов в оперативной памяти компьютера. Двоичная, шестнадцатиричная системы счисления. Форматы команд. Режимы адресации.

### Раздел 2. Элементы языка Ассемблера

#### **Тема 2.1. *Основы языка Ассемблера. Арифметические команды.***

Структура ассемблерной команды. Команды пересылки данных. Обработка целочисленных числовых данных. Арифметические команды, их разные форматы. Команды расширения целых чисел до удвоенного размера.

#### **Тема 2.2. *Типовые управляющие конструкции***

Организация конструкций ветвления в программах на языке ассемблера. Команды безусловного и условных переходов. Сравнение данных, формирование регистра флагов. Команды условной пересылки. Организация конструкций цикла. Команда loop.

#### **Тема 2.3. *Логические команды. Побитовые операции***

Логические команды. Побитовые команды. Команды сдвига.

#### **Тема 2.4. *Использование структурных данных***

Типы косвенной адресации. Работа с массивами. Строковые команды. Префиксы повторения в строковых командах.

### Раздел 3. Модульное программирование

#### **Тема 3.1. *Механизм вызова подпрограмм***

Механизм вызова подпрограмм и возврата из подпрограмм. Описание локальных переменных в подпрограммах. Встраиваемый ассемблерный код. Внешний ассемблерный код. Ассемблер и C++.

#### **Тема 3.2. *Передача параметров между модулями***

Понятие соглашения о вызовах. Наиболее распространенные соглашения о вызовах cdecl, stdcall, fastcall.

Передача параметров через стек. Передача параметров в подпрограммы через регистры.

## УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Очная форма получения высшего образования с применением  
дистанционных образовательных технологий (ДОТ)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1	2	3	4	5	6	7	8	9
<b>1</b>	<b>Принципы функционирования микропроцессоров</b>	<b>4</b>					<b>2</b>	
1.1	Архитектура компьютера. Принципиальная схема компьютера на базе процессора x86. Механизм исполнения машинных команд. Регистры процессора x86.	2						Устный опрос
1.2	Память компьютера и представление данных в памяти. Представление целочисленных данных и адресов в памяти компьютера. Системы счисления. Форматы команд. Режимы адресации.	2					2	Устный опрос
<b>2</b>	<b>Элементы языка Ассемблера</b>	<b>8</b>			<b>12</b>			
2.1	Основы языка Ассемблер. Арифметические команды. Форматы команд. Арифметические команды, команды пересылки данных.	2			2			Отчет по лабораторной работе
2.2	Типовые управляющие конструкции. Конструкции ветвления и циклов. Условные и безусловные переходы.	2			4			Отчет по лабораторной работе

2.3	Логические команды. Побитовые операции Команды сдвига.	2			2			собеседование Коллоквиум по темам 2.1-2.3.
2.4	Использование структурных данных Косвенная адресация. Работа с массивами Строковые команды	2			4			Отчет по лабораторной работе
3.	<b>Модульное программирование</b>	<b>4</b>			<b>4</b>			
3.1	Механизм вызова подпрограмм	2			2			Отчет по лабораторной работе
3.2	Передача параметров между модулями	2			2			Отчет по лабораторной работе Контрольная работа по разделу 3
	<b>ИТОГО</b>	<b>16</b>			<b>16</b>		<b>2</b>	

## ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

### Перечень основной литературы

1. Бунаков, П. Ю. Машинно-ориентированные языки программирования. Введение в ассемблер / П. Ю. Бунаков. – Санкт-Петербург : Лань, 2023. – 144 с. – Текст: электронный // Лань: электронно-библиотечная система. – URL: <https://e.lanbook.com/book/302627>.
2. Орлов, С. А. Организация ЭВМ и систем. Фундаментальный курс по архитектуре и структуре современных компьютерных средств : учебник для студ. вузов, обуч. по напр. "Информатика и вычислительная техника" / С. А. Орлов. - 4-е изд., доп. и перераб. - Санкт-Петербург [и др.] : Питер, 2018. - 687 с. - URL: <https://ibooks.ru/bookshelf/21994>.
3. Таненбаум, Э. Архитектура компьютера = Structured Computer Organization / Э. Таненбаум, Т. Остин; [пер. с англ. Е. Матвеев]. - 6-е изд. - Санкт-Петербург [и др.]: Питер, 2020. - 811 с. - URL: <https://ibooks.ru/bookshelf/361850>.

### Перечень дополнительной литературы

1. Assembler. Учебник для вузов. 2-е изд. / В.И. Юров – СПб.: Питер, 2010. – 637 с.
2. Ассемблер для процессоров Intel Pentium. / Ю.С.Магда – СПб.: Питер, 2006. – 410 с.
3. Ассемблер на примерах. Базовый курс / Рудольф Марек – СПб: Наука и Техника, 2005. – 240 с.
4. Искусство программирования на Ассемблере. Лекции и упражнения / Н.Г.Голубь – СПб.: ООО “ДиаСофтЮП”, 2002. – 656 с.
5. Бочкарева, В. В. Языки программирования низкого уровня: Практикум: учебное пособие / В. В. Бочкарева. – Москва : РТУ МИРЭА, 2021. – 83 с. – Текст: электронный // Лань : электронно-библиотечная система. – URL: <https://e.lanbook.com/book/176501>.

### Электронные ресурсы

1. Образовательный портал БГУ [Электронный ресурс]. – Режим доступа: <https://edufpmi.bsu.by/course/view.php?id=81>.
2. Образовательный портал БГУ [Электронный ресурс]. – Режим доступа: <https://edufpmi.bsu.by/course/view.php?id=750#section-4>



## **Перечень рекомендуемых средств диагностики и методика формирования итоговой отметки**

Объектом диагностики компетенций студентов являются знания, умения, полученные ими в результате изучения учебной дисциплины. Выявление учебных достижений студентов осуществляется с помощью мероприятий текущего контроля и промежуточной аттестации.

Для диагностики компетенций в рамках учебной дисциплины рекомендуется использовать следующие формы:

1. Устная форма: собеседование, устный опрос, коллоквиум.
2. Письменная форма: контрольные работы.
3. Устно-письменная форма: отчеты по лабораторным работам.

В качестве технических средств для организации работы в рамках учебной дисциплины рекомендуется использовать образовательный портал БГУ (<https://edufpmi.bsu.by>) (Moodle) - инструмент с эффективной функциональностью контроля, тренинга и самостоятельной работы.

Формой промежуточной аттестации по дисциплине «Машинно-ориентированное программирование» учебным планом предусмотрен зачет.

При формировании итоговой отметки используется рейтинговая система оценки знаний студента, дающая возможность проследить и оценить динамику процесса достижения целей обучения.

Рейтинговая система предусматривает использование весовых коэффициентов в ходе проведения контрольных мероприятий текущей аттестации.

Примерные весовые коэффициенты, определяющие вклад текущей аттестации в отметку при прохождении промежуточной аттестации:

Формирование отметки за текущую аттестацию:

- отчет по лабораторным работам – 33 %;
- контрольная работа – 34 %;
- коллоквиум – 33 %.

Итоговая отметка по дисциплине рассчитывается на основе отметки текущей аттестации (рейтинговой системы оценки знаний) - 40% и отметки на зачете - 60%.

### **Примерный перечень заданий для управляемой самостоятельной работы студентов**

В качестве заданий для управляемой самостоятельной работы могут быть выданы задания для самостоятельного решения задач по следующей теме:

**Тема 1.2. Память компьютера и представление данных в памяти.**

*Представление целочисленных данных и адресов в памяти компьютера. Системы счисления. Форматы команд. Режимы адресации. (2 ч.).*

**Примерный перечень вопросов:**

- 1) Позиционные и непозиционные системы счисления,
- 2) Перевести трехзначное десятичное число в двоичную и шестнадцатеричную систему счисления,
- 3) Представление чисел с фиксированной точкой в 1-, 2-, и 4-байтовом виде в памяти компьютера
- 4) Нумерация байтов оперативной памяти при плоской (flat) модели адресов памяти.
- 5) Минимальное и максимальное число, которое может быть представлено в 1,2 или 4 байтах.
- 6) Прямой и косвенный режим адресации операндов в командах процессора Intel32

**Форма контроля** – устный опрос.

**Примерная тематика лабораторных занятий**

Лабораторная работа 1. Системы счисления: десятичная, двоичная и шестнадцатеричная. Арифметические действия.

Лабораторная работа 2. Решение задач целочисленной арифметики.

Лабораторная работа 3. Обработка массивов.

Лабораторная работа 4. Обработка строк.

Лабораторная работа 5. Разработка модулей с использованием ассемблера.

**Рекомендуемые темы коллоквиума и контрольной работы:**

1) Коллоквиум по разделу 2 «Элементы языка ассемблера» - арифметические команды, типовые управляющие конструкции, логические команды, использование структурных данных.

2) Контрольная работа по разделу 3 «Модульное программирование».

**Примерный перечень заданий к контрольной работе**

Разработать многомодульное приложение для решения следующей задачи:

1. Ввести массив целых чисел. Ввод организовать из головного модуля на C++. Сформировать массив, состоящий из чисел, которые являются заданной степенью 2. (Ассемблер). Вывести результат (C++).

2. Ввести целое число. Ввод числа организовать из головного модуля на C++. Вычислить сумму цифр модуля этого числа (Ассемблер). Вывести результат (C++).

3. Ввести целое число. Ввод числа организовать из головного модуля на C++. Определить, является ли модуль этого числа квадратом некоторого числа, если «да», то найти его (Ассемблер). Вывести результат (C++).

4. Ввести целое число. Ввод числа организовать из головного модуля на C++. Определить количество четных цифр числа (Ассемблер). Вывести результат (C++).

5. Ввести массив целых чисел. Ввод организовать из головного модуля на C++. Определить целую часть среднего арифметического чисел массива (Ассемблер). Вывести результат (C++).

6. Ввести два целых положительных числа. Ввод числа организовать из головного модуля на C++. Определить среднее арифметическое сумм их цифр (Ассемблер). Вывести результат (C++).

7. Ввести целое число. Ввод числа организовать из головного модуля на C++. Определить среднее арифметическое наибольшей и наименьшей цифр в записи числа (Ассемблер). Вывести результат (C++).

### **Описание инновационных подходов и методов к преподаванию учебной дисциплины**

При организации образовательного процесса используются следующие методы:

- *метод группового обучения*, который представляет собой форму организации учебно-познавательной деятельности обучающихся, предполагающую функционирование разных типов малых групп, работающих как над общими, так и специфическими учебными заданиями.

В качестве технических средств для организации работы в рамках учебной дисциплины рекомендуется использовать образовательный портал БГУ (<https://edufpmi.bsu.by>) – инструмент с эффективной функциональностью контроля, тренинга и самостоятельной работы.

### **Методические рекомендации по организации самостоятельной работы обучающихся**

Для организации самостоятельной работы студентов по учебной дисциплине следует использовать современные информационные ресурсы: разместить на образовательном портале комплекс учебных и учебно-методических материалов (учебно-программные материалы, учебное издание для теоретического изучения дисциплины, методические указания к лабораторным занятиям, материалы текущего контроля и промежуточной аттестации, позволяющие определить соответствие учебной деятельности обучающихся требованиям образовательного стандарта общего высшего образования и учебно-программной документации, в том числе вопросы для подготовки к зачету, задания, тесты, вопросы для самоконтроля и др., список рекомендуемой литературы, информационных ресурсов и др.).

Задания УСР по учебной дисциплине рекомендуется делить на три модуля:

1. Задания, формирующие достаточные знания по изученному учебному материалу **на уровне узнавания**:
  - a) Опишите механизм сравнения целых чисел, знаковых и беззнаковых. Какие флаги при этом формируются? Какие операторы условного перехода используются в каждом случае?
  - b) Перечислите несколько способов организации итерационных циклов (аналог цикла for в языке программирования C++).
  - c) Найдите количество сочетаний из n элементов по m.
2. Задания, формирующие компетенции **на уровне воспроизведения**:
  - a) Определить, является ли натуральное число симметричным.
  - b) Вычислить количество значащих разрядов в двоичной записи натурального числа.
  - c) Перевести число в двоичное представление.
  - d) Найти медианную цифру числа.
3. Задания, формирующие компетенции **на уровне применения полученных знаний**:
  - a) Определить количество локальных минимумов в массиве целых чисел.
  - b) Подсчитать количество чисел в массиве, равных заданному.
  - c) Разложить целое число на простые множители.
  - d) Найти все множители заданного числа.

### **Примерный перечень вопросов к зачету**

1. Регистры процессора. Типы регистров.
2. Регистры общего назначения.
3. Регистр флагов. Перечислить известные флаги.
4. Работа со стеком. Команды работы со стеком.
5. Представление целых чисел в двоичной и шестнадцатеричной системах счисления.
6. Команды сложения и вычитания.
7. Операторы преобразования длины.
8. Команды пересылки данных.
9. Команды для реализации конструкции разветвления
10. Команды для организации цикла.
11. Команды условных переходов.
12. Организация циклов с помощью переходов.
13. Организация циклов с помощью команды LOOP и ее модификаций.
14. Команды умножения целых чисел.
15. Команды деления целых чисел.
16. Команды для работы с битами.
17. Команды линейного сдвига.
18. Команды циклического сдвига.
19. Доступ к элементам одномерных массивов.
20. Строковые команды. Префиксы повторения в строковых командах.

21. Механизм вызова процедур. Команды для работы с процедурами.
22. Способы передачи параметров в процедуры.
23. Организация работы с локальными переменными.
24. Вызов ассемблерных процедур из программ на C++.
25. Соглашения о вызовах процедур.



## ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УВО

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Разработка кросс-платформенных приложений	Технологий программирования	Нет	Изменений не требуется (протокол № 16 от 18 мая 2023 г.)
Операционные системы	Технологий программирования	Нет	Изменений не требуется (протокол № 16 от 18 мая 2023 г.)
Промышленное программирование	Технологий программирования	Нет	Изменений не требуется (протокол № 16 от 18 мая 2023 г.)..
Технологии программирования	Технологий программирования	Нет	Изменений не требуется (протокол № 16 от 18 мая 2023 г. )

**ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ**  
на \_\_\_\_ / \_\_\_\_ учебный год

№№ Пп	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры технологий программирования (протокол № \_ \_ г.)

Заведующий кафедрой

\_\_\_\_\_ профессор д.т.н.\_\_\_\_  
(ученая степень, звание)

\_\_\_\_\_  
(подпись)

\_\_\_\_\_ А.Н. Курбацкий  
(И.О. Фамилия)

УТВЕРЖДАЮ

Декан факультета

\_\_\_\_\_  
(ученая степень, звание)

\_\_\_\_\_  
(подпись)

\_\_\_\_\_  
(И.О. Фамилия)