

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ  
Заведующий кафедрой  
Математического моделирования



Бурлуцкая М.Ш.

16.04.2024 г.

**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ ПРОГРАММЫ  
ПОДГОТОВКИ СПЕЦИАЛИСТОВ СРЕДНЕГО ЗВЕНА**

*ОП.05 Основы алгоритмизации и программирования*

*Код и наименование дисциплины в соответствии с Учебным планом*

*09.02.07 Информационные системы и программирование*

*Код и наименование специальности*

*технический*

*Профиль подготовки (технический, естественнонаучный, социально-экономический,  
гуманитарный)*

*специалист по информационным системам*

*Квалификация выпускника*

*очная*

*Форма обучения*

Учебный год: 2024-2025

Семестр(ы): 1-2

Рекомендована: Научно-методическим советом математического факультета  
протокол от 28.03.2024 № 0500-03

Составители программы: Волобуева Софья Алексеевна, преподаватель кафедры  
математического моделирования

2024 г.

## СОДЕРЖАНИЕ

1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

# 1. ПАСПОРТ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

## ОП.05 Основы алгоритмизации и программирования

Программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта среднего профессионального образования (ФГОС СПО) по специальности 09.02.07 «Информационные системы и программирование», утвержденного приказом Министерства образования и науки от 9 декабря 2016 года № 1547 (зарегистрирован Министерством юстиции Российской Федерации 26 декабря 2016г., регистрационный №44936), входящей в укрупненную группу специальностей (09.00.00) Информатика и вычислительная техника (с изменениями и дополнениями от 17 декабря 2020 г., 1 сентября 2022 г.).

### 1.1. Область применения программы

Рабочая программа учебной дисциплины является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности СПО 09.02.07 «Информационные системы и программирование» (базовой подготовки).

### 1.2. Место дисциплины в структуре основной профессиональной образовательной программы

Учебная дисциплина является общепрофессиональной дисциплиной профессионального цикла.

### 1.3. Цели и задачи дисциплины – требования к результатам освоения дисциплины:

В результате освоения дисциплины обучающийся должен **уметь**:

- Разрабатывать алгоритмы для конкретных задач.
- Использовать программы для графического отображения алгоритмов.
- Определять сложность работы алгоритмов.
- Работать в среде программирования.
- Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.
- Оформлять код программы в соответствии со стандартом кодирования.
- Выполнять проверку, отладку кода программы.

В результате освоения дисциплины обучающийся должен **знать**:

- Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.
- Эволюцию языков программирования, их классификацию, понятие системы программирования.
- Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.
- Подпрограммы, составление библиотек подпрограмм.
- Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения.
- Интегрированные среды разработчика.

Результатом освоения программы учебной дисциплины является овладение обучающимся профессиональными (ПК) и общими (ОК) компетенциями:

<b>Код</b>	<b>Формулировка компетенции</b>
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
ПК 2.4	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения
ПК 2.5	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования
ПК 3.1	Осуществлять ревьюирование программного кода в соответствии с технической документацией

#### **1.4. Рекомендуемое количество часов на освоение программы дисциплины:**

максимальной учебной нагрузки обучающегося 174 часов, в том числе:  
аудиторной учебной работы обучающегося (обязательных учебных занятий) 134 часов;  
внеаудиторной (самостоятельной) учебной работы обучающегося 28 часов.

## **2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ**

### **2.1. Объем учебной дисциплины и виды учебной работы**

<b>Вид учебной деятельности</b>	<b>Объем часов</b>
<b>Максимальная учебная нагрузка (всего)</b>	<b>174</b>
<b>Обязательная аудиторная учебная нагрузка (всего)</b>	<b>134</b>
в том числе:	
лекции	66
лабораторные работы	66
практические занятия	
контрольные работы	*
курсовая работа (проект) (если предусмотрено)	*
<b>Самостоятельная работа обучающегося (всего)</b>	<b>28</b>
<b>Итоговая аттестация в форме</b>	<b>Экзамен 12</b>



**Тематический план и содержание учебной дисциплины ОП.05 «Основы алгоритмизации и программирования»**

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающегося, курсовая работа (проект)	Объем в часах	Уровень освоения
1	2	3	4
<b>Раздел 1.</b>	<b>Введение в программирование</b>		2
<b>Тема 1.1. Языки программирования</b>	<b>Содержание учебного материала</b>	6	
	1. Развитие языков программирования.		
	2. Обзор языков программирования. Области применения языков программирования. Стандарты языков программирования. Среда проектирования. Компиляторы и интерпретаторы.		
	3. Жизненный цикл программы. Программа. Программный продукт и его характеристики.		
	4. Основные этапы решения задач на компьютере.		
	<b>В том числе практических занятий и лабораторных работ Самостоятельная работа обучающихся</b>		
<b>Тема 1.2. Типы данных</b>	<b>Содержание учебного материала</b>	4	
	1. Типы данных. Простые типы данных. Производные типы данных. Структурированные типы данных.		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Раздел 2.</b>	<b>Содержание учебного материала</b>	<b>40</b>	2
<b>Тема 2.1. Операторы языка программирования</b>	1. Операции и выражения. Правила формирования и вычисления выражений. Структура программы. Ввод и вывод данных. Оператор присваивания. Составной оператор.		
	2. Условный оператор. Оператор выбора.		
	3. Цикл с постусловием. Цикл с предусловием. Цикл с параметром. Вложенные циклы.		
	4. Массивы. Двумерные массивы. Строки. Стандартные процедуры и функции для работы со строками.		
	5. Структурированный тип данных – множество. Операции над множествами.		
	6. Комбинированный тип данных – запись. Файлы последовательного доступа. Файлы прямого доступа		
	<b>В том числе практических занятий и лабораторных работ Самостоятельная работа обучающихся</b>		
<b>Раздел 3.</b>	<b>Содержание учебного материала</b>		2
<b>Тема 3.1. Процедуры и функции</b>	1. Общие сведения о подпрограммах. Определение и вызов подпрограмм. Область видимости и время жизни переменной. Механизм передачи параметров. Организация функций.	12	
	2. Рекурсия. Программирование рекурсивных алгоритмов.		

	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Тема 3.2. Структуризация в программировании</b>	<b>Содержание учебного материала</b>	<b>4</b>	
	1. Основы структурного программирования. Методы структурного программирования.		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Тема 3.3. Модульное программирование</b>	<b>Содержание учебного материала</b>	<b>12</b>	
	1. Модульное программирование. Понятие модуля. Структура модуля. Компиляция и компоновка программы.		
	2. Стандартные модули.		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Раздел 4</b>	<b><i>Основные конструкции языков программирования</i></b>		
<b>Тема 4.1 Указатели.</b>	<b>Содержание учебного материала</b>	<b>12</b>	2
	1. Указатели. Описание указателей. Основные понятия и применение динамически распределяемой памяти. Создание и удаление динамических переменных.		
	2. Структуры данных на основе указателей.		
	3. Задача о стеке.		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Раздел 5</b>	<b>Содержание учебного материала</b>		
<b>Тема 5.1 Основные принципы объектно-ориентированного программирования (ООП)</b>	1. История развития ООП. Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс.	<b>12</b>	
	2. Основные принципы ООП: инкапсуляция, наследование, полиморфизм.		
	3. Классы объектов. Компоненты и их свойства.		
	4. Событийно-управляемая модель программирования. Компонентно-ориентированный подход.		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Тема 5.2 Интегрированная среда разработчика.</b>	<b>Содержание учебного материала</b>	<b>12</b>	2
	1. Требования к аппаратным и программным средствам интегрированной среды разработчика.		
	2. Интерфейс среды разработчика: характеристика, основные окна, инструменты, объекты. Форма и размещение на ней управляющих элементов.		
	3. Панель компонентов и их свойства. Окно кода проекта.		
	4. Состав и характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.		
	5. Панель компонентов и их свойства. Окно кода проекта. Состав и		

	характеристика проекта. Выполнение проекта. Настройка среды и параметров проекта.		
	6. Настройка среды и параметров проекта.		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Тема 5.3. Визуальное событийно-управляемое программирование</b>	<b>Содержание учебного материала</b>	<b>10</b>	
	1. Основные компоненты (элементы управления) интегрированной среды разработки, их состав и назначение.		
	2. Дополнительные элементы управления. Свойства компонентов. Виды свойств. Синтаксис определения свойств. Назначения свойств и их влияние на результат. Управление объектом через свойства.		
	3. События компонентов (элементов управления), их сущность и назначение. Создание процедур на основе событий.		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Тема 5.4 Разработка оконного приложения</b>	<b>Содержание учебного материала</b>	<b>12</b>	
	1. Разработка функционального интерфейса приложения. Создание интерфейса приложения.		
	2. Разработка функциональной схемы работы приложения.		
	3. Разработка игрового приложения.		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Тема 5.5 Этапы разработки приложений</b>	<b>Содержание учебного материала</b>	<b>10</b>	
	1. Разработка приложения.		
	2. Проектирование объектно-ориентированного приложения.		
	3. Создание интерфейса пользователя.		
	4. Тестирование, отладка приложения.		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Тема 5.6 Иерархия классов.</b>	<b>Содержание учебного материала</b>	<b>12</b>	
	1. Классы ООП: виды, назначение, свойства, методы, события.		
	2. Перегрузка методов.		
	3. Тестирование и отладка приложения.		
	4. Решение задач		
	<b>В том числе практических занятий и лабораторных работ</b>		
	<b>Самостоятельная работа обучающихся</b>		
<b>Примерная тематика практических занятий и лабораторных работ:</b>			
Знакомство со средой программирования.			
Составление программ линейной структуры.			
Составление программ разветвляющейся структуры.			
Составление программ циклической структуры			
Обработка одномерных массивов.			



<p>Обработка двумерных массивов.  Работа со строками.  Работа с данными типа множество.  Файлы последовательного доступа.  Типизированные файлы.  Нетипизированные файлы.  Организация процедур.  Организация функций.  Применение рекурсивных функций.  Программирование модуля.  Создание библиотеки подпрограмм.  Использование указателей для организации связанных списков.  Изучение интегрированной среды разработчика.  Создание проекта с использованием компонентов для работы с текстом.  Создание проекта с использованием компонентов ввода и отображения чисел, дат и времени.  События компонентов (элементов управления), их сущность и назначение.  Создание процедур на основе событий.  Создание проекта с использованием кнопочных компонентов.  Создание проекта с использованием компонентов стандартных диалогов и системы меню.  Разработка функциональной схемы работы приложения.  Разработка оконного приложения с несколькими формами.  Разработка игрового приложения.  Создание процедур обработки событий. Компиляция и запуск приложения.  Разработка интерфейса приложения.  Тестирование, отладка приложения.  Классы ООП: виды, назначение, свойства, методы, события.  Объявления класса.  Создание наследованного класса.  Программирование приложений.  Перегрузка методов.</p>		
<b>Промежуточная аттестация</b>	<b>12</b>	
<b>Всего:</b>	<b>174</b>	

Для характеристики уровня освоения учебного материала используются следующие обозначения:

1. – ознакомительный (узнавание ранее изученных объектов, свойств);
2. – репродуктивный (выполнение деятельности по образцу, инструкции или под руководством)
3. – продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач)

### **3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ДИСЦИПЛИНЫ**

#### **3.1. Материально-техническое обеспечение**

Лаборатория «Программирования и баз данных» (ауд.40/4):  
автоматизированные рабочие места для обучающихся и преподавателя, специализированная мебель для сервисного обслуживания ПК с заземлением и защитой от статического напряжения, проектор и экран, виртуальный сервер, маркерная доска.  
Microsoft Windows 10 Pro 64 bit, Xubuntu 20.4, Microsoft Visual Studio 2019 (Community), MySQL Server Community, MySQL Shell Community, MySQL WorkBench Community, MySQL Connector Net, Lazarus, Free Pascal, Java, JRE, Deductor (Academic), Loginom, MatLab, NetBeans IDE, Python 2/3, Maxima, WinDjView, Wing-101, Foxit Reader, 7-Zip (Архиватор), Mozilla Firefox, Oracle VirtualBox, VMware-player

#### **3.2. Информационное обеспечение обучения**

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

Основные источники:

1. Окулов, С.М. Динамическое программирование : учебное пособие : [12+] / С.М. Окулов, О.А. Пестов. – 3-е изд., электрон. – Москва : Лаборатория знаний, 2020. – 299 с. – (Развитие интеллекта школьников). – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=445917> – ISBN 978-5-00101-683-0.
2. Нагаева, И.А. Алгоритмизация и программирование. Практикум : учебное пособие : [12+] / И.А. Нагаева, И.А. Кузнецов. – Москва ; Берлин : Директ-Медиа, 2019. – 168 с. : ил., табл. – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=570287> – Библиогр.: с. 163-164. – ISBN 978-5-4499-0314-3.
3. Окулов, С.М. Основы программирования : учебное пособие : [16+] / С.М. Окулов. – 10-е изд., электрон. – Москва : Лаборатория знаний, 2020. – 339 с. – (Развитие интеллекта школьников). – Режим доступа: по подписке. – URL: <http://biblioclub.ru/index.php?page=book&id=221981> – ISBN 978-5-00101-759-2.
4. Быкадорова, Е. А. Программирование. Практикум : учебное пособие / Е. А. Быкадорова. — Санкт-Петербург : Лань, 2020. — 60 с. — ISBN 978-5-8114-4612-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/139323>
5. Быкадорова, Е. А. Основы программирования информационного контента : учебное пособие / Е. А. Быкадорова, О. Н. Синявская. — Санкт-Петербург : Лань, 2020. — 72 с. — ISBN 978-5-8114-4567-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL:

---

Дополнительные источники:

6. Таненбаум, Эндрю. Современные операционные системы = Modern Operating Systems / Э. Таненбаум ; [пер. с англ. Н. Вильчинского, А. Лашкевича] .— 3-е изд. — СПб. [и др.] : Питер, 2012 .— 1115 с.
7. Ульман, Ларри. Основы программирования на PHP : Самоучитель: Пер. с англ. / Ларри Ульман .— М. : ДМК Пресс, 2001 .— 285 с.
8. Богачев, Кирилл Юрьевич. Основы параллельного программирования : [Учебное пособие] / К.Ю. Богачев .— М. : Бином. Лаборатория знаний, 2003 .— 342 с.
9. Окулов, Станислав Михайлович. Основы программирования : [учебное пособие] / С. Окулов .— 2-е изд., испр. — М. : БИНОМ. Лаборатория знаний, 2005
10. Основы языка программирования С++ с применением объектно-ориентированного программирования : учебно-методическое пособие для вузов / Воронеж. гос. ун-т; сост. М.К. Чернышов .— Воронеж : ИПЦ ВГУ, 2007 .— 68 с. — Библиогр.: с.67 .— <URL:<http://www.lib.vsu.ru/elib/texts/method/vsu/m07-181.pdf>>.
11. Окулов, С.М. Программирование в алгоритмах : учебное пособие : [16+] / С.М. Окулов. — 6-е изд. (эл.). — Москва : Лаборатория знаний, 2017. — 386 с. — (Развитие интеллекта школьников). — Режим доступа: по подписке. — URL: <http://biblioclub.ru/index.php?page=book&id=461264> — Библиогр. в кн. — ISBN 978-5-00101-449-2.

---

**Интернет-ресурсы:**

Электронный каталог Зональной научной библиотеки ВГУ (<http://www.lib.vsu.ru>)

---

#### **4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в процессе проведения практических занятий и лабораторных работ, контрольных работ.

#### **Критерии оценки результата итогового контроля по итогам освоения дисциплины:**

**Отлично:** выполнены все задания, грамотно и логично изложен ответ на практико-ориентированные вопросы, обоснованы высказывания с точки зрения известных теоретических положений.

**Хорошо:** если обучающийся полно освоил учебный материал, владеет научно-понятийным аппаратом, ориентируется в изученном материале, осознанно применяет знания на практике, грамотно излагает ответ, но содержание и форма ответа имеют отдельные неточности.

**Удовлетворительно:** если обучающийся обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, в применении теоретических знаний при ответе на практико-ориентированные письменные задания; не умеет доказательно обосновать свои суждения.

**Неудовлетворительно:** если студент имеет разрозненные, бессистемные знания по дисциплине, допускает ошибки в определении базовых понятий, искажает их смысл; не может практически применять теоретические знания.

<b>Результаты обучения (освоенные умения, усвоенные знания)</b>	<b>Формы и методы контроля и оценки результатов обучения</b>
<p><b>Умения:</b></p> <p>Разрабатывать алгоритмы для конкретных задач. Использовать программы для графического отображения алгоритмов. Определять сложность работы алгоритмов. Работать в среде программирования. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. Оформлять код программы в соответствии со стандартом кодирования. Выполнять проверку, отладку кода программы.</p>	<p>Умеет разрабатывать алгоритмы для конкретных задач, использовать программы для графического отображения алгоритмов, определять сложность работы алгоритмов, работать в среде программирования, реализовывать построенные алгоритмы в виде программ на конкретном языке программирования, оформлять код программы в соответствии со стандартом кодирования, а также выполнять проверку, отладку кода программы.</p>
<p><b>Знания:</b></p> <p>Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. Эволюцию языков программирования, их классификацию, понятие системы программирования. Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти. Подпрограммы, составление библиотек подпрограмм Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения Интегрированные среды разработчика.</p>	<p>Знает понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции, эволюцию языков программирования, их классификацию, понятие системы программирования, основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти, подпрограммы, составление библиотек подпрограмм, объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения,</p>

	интегрированные среды разработчика
--	------------------------------------

<b>Результаты обучения</b>	<b>Основные показатели оценки результата</b>
ОК 02	Использует современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
ПК 2.4	Осуществляет разработку тестовых наборов и тестовых сценариев для программного обеспечения
ПК 2.5	Производит инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования
ПК 3.1	Осуществляет ревьюирование программного кода в соответствии с технической документацией