

Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет»

" УТВЕРЖДАЮ

Первый проректор Белорусско-
Российского университета

Ю. В. Машин

20.10 2023

Регистрационный № УД-150306/Б.Р.В.131

ПРОГРАММИРОВАНИЕ И ОСНОВЫ АЛГОРИТМИЗАЦИИ
(наименование дисциплины)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Направление подготовки 15.03.06 Мехатроника и робототехника

Направленность (профиль) Робототехника и робототехнические системы: разработка и применение

Квалификация Бакалавр

	Форма обучения
	Очная
Курс	2
Семестр	4
Лекции, часы	34
Практические занятия, часы	16
Лабораторные занятия, часы	16
Зачёт, семестр	4
Контактная работа по учебным занятиям, часы	66
Самостоятельная работа, часы	78
Всего часов / зачетных единиц	144/4

Кафедра-разработчик программы: Программное обеспечение информационных технологий

Составитель: канд.техн.наук, доц. Н.Н. Горбатенко
(И.О. Фамилия, ученая степень, ученое звание)

Могилев, 2023

Рабочая программа составлена в соответствии с Федеральным Государственным образовательным стандартом высшего профессионального образования по направлению подготовки 15.03.06 «Мехатроника и робототехника», утвержденным приказом № 1046 от 17.08.2020 г., учебным планом рег. № 150306-2.1, утвержденным 28.04.2023

Рассмотрена и рекомендована к утверждению кафедрой ПОИТ
20.09.2023 г., протокол № 2.

Зав. кафедрой  В.В. Кутузов

Одобрена и рекомендована к утверждению Научно-методическим советом
Белорусско-Российского университета

«18» 10 2023, протокол № 2.

Зам. председателя
Научно-методического совета

 С.А. Сухоцкий

Рецензент:

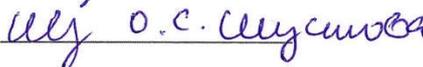
Акиншева И. В., заведующий кафедрой программного обеспечения информационных технологий МГУ имени А. А. Кулешова, канд. техн. наук

(И.О. Фамилия, должность, ученая степень, ученое звание рецензента)

Рабочая программа согласована
Зав. каф. ТМ

 В.М. Шеменков

Ведущий библиотекарь

 О.С. Шущкова

Начальник учебно-методического
отдела

 О.Е. Печковская

1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1 Цель учебной дисциплины

Цель преподавания дисциплины – ознакомить студентов с базовыми понятиями программирования, методами построения алгоритмов и разработки компьютерных программ; на примере языка C/C++ изложить основные концепции алгоритмических языков программирования.

1.2 Задачи учебной дисциплины

В результате изучения учебной дисциплины обучаемый должен:

знать:

- современное состояние языка программирования C/C++;
- основные динамические структуры данных и алгоритмы их обработки;
- вычислительные алгоритмы решения инженерных задач;
- теоретические основы алгоритмизации и проектирования программ;

уметь:

- выполнять алгоритмизацию инженерных задач;
- реализовывать разработанный алгоритм в виде собственной программы на алгоритмическом языке или с использованием стандартных программ;
- применять разработанные программы в профессиональной деятельности;

владеть:

- современными средствами программирования;
- навыками анализа исходных и выходных данных решаемых задач и формами их представления;
- навыками отладки программ.

1.3 Место дисциплины в системе подготовки студента

Дисциплина относится к «Часть Блока1, формируемая участниками образовательных отношений. Элективные дисциплины».

Перечень учебных дисциплин, которые будут опираться на данную дисциплину:

- Программное обеспечение мехатронных и робототехнических систем;
- Микропроцессорная техника в мехатронике и робототехнике.

Кроме того, знания, полученные при изучении дисциплины на лекциях и лабораторных работах будут применены при прохождении учебной и производственных практик, а также при подготовке выпускной квалификационной работы и дальнейшей профессиональной деятельности.

1.4 Требования к освоению учебной дисциплины

Освоение данной учебной дисциплины должно обеспечивать формирование следующих компетенций:

Коды формируемых компетенций	Наименования формируемых компетенций
ПК-7	Способен проводить конструкторские и расчетные работы по проектированию гибких производственных систем в машиностроении

2 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Вклад дисциплины в формирование результатов обучения выпускника (компетенций) и достижение обобщенных результатов обучения происходит путём освоения содержания обучения и достижения частных результатов обучения, описанных в данном разделе.

2.1 Содержание учебной дисциплины

Номер тем	Наименование тем	Содержание	Коды формируемых компетенций
1	Введение в язык C++	<p>Структура программы. Функция main(). Директивы препроцессора. Инструкции. Виды инструкций: объявления, выражения, управляющие конструкции. Встроенные функции. Функции пользователя. Комментарии.</p> <p>Ввод и вывод в консоль. Директива using. Подключение пространств имен и определение псевдонимов.</p> <p>Объявления переменных. Фундаментальные типы: int, long, long long, char, float, double, long double, bool. Модификаторы типов. Идентификаторы. Литералы. Неявные и явные преобразования типов. Область видимости переменных.</p> <p>Выражения. Базовые операторы. Арифметические операторы. Операторы сравнения. Побитовые операторы. Логические операторы. Операторы присваивания. Операторы инкремента и декремента. Тернарный оператор. Оператор sizeof(). Оператор запятая. Приоритет операторов. Переопределение порядка выполнения операторов в выражении.</p>	ПК-7
2	Управляющие инструкции	<p>Инструкция if. Условные выражения. Вложенные инструкции if. Инструкция if-else-if.</p> <p>Инструкция switch.</p> <p>Цикл for. Варианты цикла for.</p> <p>Цикл while.</p> <p>Цикл do-while.</p> <p>Объявление переменных в условных инструкциях и циклах.</p> <p>Инструкции перехода: continue; break; goto.</p> <p>Вложенные циклы.</p>	ПК-7
3	Компиляция и запуск программы	<p>Назначение компиляции. Этапы компиляции. Препроцессинг. Ассемблирование. Компилирование. Линковка.</p> <p>Основы препроцессорной обработки. Директивы препроцессора. #include, #define, #if, #ifdef, #ifndef, #else, #elif, #endif, #error, #pragma once.</p> <p>Виды ошибок при запуске программы. Ошибки времени компиляции (Compile-time errors). Ошибки времени выполнения (runtime errors). Неопределенное поведение (undefined behavior).</p>	ПК-7
4	Массивы. Строки. Перечисления. Структуры.	<p>Понятие массива. Виды массивов.</p> <p>Одномерные массивы. Объявление. Инициализация. Доступ к элементам массива. Получение длины массива. Перебор элементов массива с помощью цикла for. Ввод/вывод значений массива в консоль.</p> <p>Двумерные массивы. Объявление. Инициализация. Доступ к элементам массива. Перебор элементов</p>	ПК-7

		<p>массива с помощью цикла <code>for</code>. Ввод/вывод значений массива в консоль.</p> <p>Выход индекса массива за пределы допустимого диапазона.</p> <p>Одномерные массивы символов. Двумерные массивы символов.</p> <p>Перебор двумерных символьных массивов в цикле <code>for</code>.</p> <p>Ввод символьных массивов с консоли.</p> <p>Введение в строки. Недостатки строк в C-стиле с завершающим нулевым символом.</p> <p>Использование класса <code>std::string</code> из стандартной библиотеки <code><string></code> для работы со строками. Создание и инициализация переменных типа <code>string</code>. Получение и изменение символов строки. Доступ к символам строки по индексу. Чтение строки с консоли.</p> <p>Пользовательские типы данных: перечисления, структуры. Объявление и использование перечислений. Объявление и определение структур. Инициализация структур. Присваивание значений членам структур. Массивы структур.</p>	
5	Ссылки и указатели	<p>Понятие ссылки. Формат объявления ссылки. Ссылки на переменные и константы. Область применения ссылок.</p> <p>Понятие указателя. Определение указателя. Получение адреса переменной с помощью оператора <code>&</code>. Инициализация указателя. Нулевой указатель <code>nullptr</code>. Получение значения переменной по ее адресу. Операция разыменованье указателя <code>*</code>.</p> <p>Операции с указателями. Присваивание адреса. Разыменованье указателя. Присвоение указателю другого указателя. Ссылки на указатели. Адрес указателя. Операции сравнения. Приведение типов.</p> <p>Арифметические операции над указателями: сложение, вычитание, инкремент, декремент.</p> <p>Указатели и массивы. Указатель на строки и массивы символов. Массивы указателей.</p>	ПК-7
6	Память программы	<p>Структура памяти программы на языке C++ в ОЗУ компьютера: область кода программы, область данных (статическая память), область стека (автоматическая память).</p> <p>Область кода программы как место хранения программы в машинных кодах.</p> <p>Статическая память как область памяти, в которой хранится информация на протяжении всего времени работы программы (глобальные переменные, локальные статические переменные).</p> <p>Стековая память как область памяти для хранения локальных переменных, объявленных в функции. Работа стека. Указатель стека. Адрес возврата функции. Размер стека. Переполнение стека.</p> <p>Динамическая память. Выделение памяти. Освобождение памяти. Оператор <code>new</code>. Оператор <code>delete</code>. Выделение и освобождение динамической памяти под простую переменную и массив. Динамические одномерные и многомерные массивы.</p> <p>Ошибки, возникающие при работе с динамической памятью.</p> <p>Использование класса <code>std::vector</code> из стандартной библиотеки <code><vector></code> для реализации динамических массивов. Создание объекта <code>vector</code>. Инициализация вектора. Длина вектора. Емкость вектора. Доступ к элементам вектора. Проход по элементам вектора. Ввод/вывод элементов вектора в консоль.</p>	ПК-7

7	Функции	<p>Определение функции. Объявление функции. Выполнение функции.</p> <p>Область видимости и время жизни переменных. Глобальные переменные. Локальные переменные. Автоматические объекты. Скрытие переменных. Статические переменные.</p> <p>Параметры функции. Аргументы функции. Аргументы по умолчанию. Автоматическое выведение типа параметров.</p> <p>Передача аргументов по значению и по ссылке. Передача аргументов по значению. Передача параметров по ссылке. Преобразования типов.</p> <p>Константные параметры. Константные ссылки.</p> <p>Оператор return и возвращение результата. return без возвращения значения. Выведение типа результата.</p> <p>Указатели в параметрах функции. Константные параметры-указатели. Параметры по ссылке или параметры-указатели.</p> <p>Массивы в параметрах функции. Передача маркера конца массива. Константные массивы. Передача массива по ссылке. Передача многомерного массива.</p>	ПК-7
8	Многомодульные программы	<p>Понятие модуля как функционально законченного фрагмента программы, оформленного в виде отдельного файла с исходным кодом, предназначенным для использования в других модулях.</p> <p>Преимущества использования многомодульных программ.</p> <p>Структура файлов многомодульных программ: главный файл, файл заголовков, файл реализации.</p> <p>Главный файл. Файл с расширением .cpp, содержащий метод main()</p> <p>Файл заголовков. Файл с расширением .h, который содержит объявления пользовательских типов.</p> <p>Файл реализации. Файл с расширением .cpp, который содержит определения пользовательских типов объявленных в файле заголовков.</p> <p>Способы защиты от повторного включения заголовочных файлов в программу.</p>	ПК-7
9	Принципы объектно-ориентированного программирования	<p>Парадигмы программирования: процедурная; объектно-ориентированная; обобщенная; функциональная.</p> <p>Объектно-ориентированное программирование: сущность; отличие от процедурной парадигмы программирования; достоинства; недостатки; область применения.</p> <p>Основные принципы объектно-ориентированного программирования. Инкапсуляция. Наследование. Полиморфизм.</p> <p>Объектная декомпозиция задачи.</p> <p>Абстрагирование.</p>	ПК-7
10	Классы и объекты	<p>Понятие класса и объекта класса. Формат объявления класса. Обзор членов класса. Члены-данные (поля класса). Члены-функции (методы класса). Члены-типы (вложенные пользовательские типы). Область видимости класса. Спецификаторы public, private, protected.</p> <p>Создание объекта класса. Указатели на объекты.</p> <p>Действия над объектами классов.</p> <p>Сущность инкапсуляции. Механизм реализации инкапсуляции. Закрытые поля класса. Открытые интерфейсные методы класса. Геттеры. Сеттеры. Преимущества инкапсуляции.</p> <p>Объявление и определение функций класса.</p>	ПК-7

		Встроенные функции. Ключевое слово <code>inline</code> . Раздельное объявление и определение функций класса. Скрытый указатель <code>this</code> .	
11	Специальные методы класса	Конструкторы. Виды конструкторов: конструктор умолчания, конструктор копирования, конструктор преобразования. Примеры конструкторов. Деструктор. Правила автоматической генерации специальных методов класса: конструктора умолчания, конструктора копирования, деструктора, операторного метода присваивания. Поверхностное копирование. Глубокое копирование. Порядок вызова конструкторов и деструкторов.	ПК-7
12	Статические члены класса. Дружественные функции и классы	Особенности статических членов-данных и членов функций класса. Особенности использования статических методов класса. Друзья класса. Виды друзей класса: обычная функция, функция-член другого класса, весь класс. Свойства друзей класса. Преимущества использования друзей класса.	ПК-7
13	Перегрузка операций	Понятие перегрузки операций над объектами класса. Операторная функция. Формат объявления операторной функции. Способы перегрузки операций: с помощью метода класса, с помощью внешней функции, функции-друга. Пример перегрузки операции '+' двух комплексных чисел с помощью функции-члена класса и функции друга. Пример перегрузка операции '*' комплексного числа на вещественное число.	ПК-7
14	Наследование классов	Понятие наследования классов. Терминология (родительский класс, дочерний класс). Виды наследования: одиночное наследование, множественное наследование. Одиночное наследование. Формат наследования. Порядок построения дочерних классов. Конструкторы и инициализация. Инициализация членов родительского класса. Цепочки наследований. Деструкторы. Спецификатор доступа <code>protected</code> . Типы наследований. Наследование типа <code>public</code> . Наследование типа <code>private</code> . Наследование типа <code>protected</code> . Добавление нового функционала в дочерний класс. Вызов методов родительского класса. Переопределение методов родительского класса. Скрытие методов родительского класса. Множественное наследование. Проблемы с множественным наследованием.	ПК-7
15	Виртуальные функции и полиморфизм	Понятие полиморфизма. Виды полиморфизма: статический, динамический. Понятие виртуальной функции. Наследование виртуальных функций. Модификаторы <code>virtual</code> и <code>override</code> . Чистые виртуальные функции. Интерфейсы. Абстрактные функции. Повышающее и понижающее приведение типов в иерархии классов. Повышающее преобразование и использование оператора <code>dynamic_cast</code> . Понижающее приведение и использование оператора <code>static_cast</code> .	ПК-7
16	Потоки ввода и вывода	Библиотека <code>iostream</code> . Потоки в C++. Ввод/вывод в	ПК-7

		<p>C++. Стандартные потоки в C++.</p> <p>Класс istream. Оператор извлечения.</p> <p>Классы ostream и ios. Форматирование вывода. Флаги, манипуляторы и методы форматирования. Точность, запись чисел и десятичная точка. Ширина поля, символы-заполнители и выравнивание.</p> <p>Режимы открытия файлов. Явное открытие файлов с помощью функции open(). Чтение и запись текстовых файлов. Чтение и запись двоичных файлов.</p>	
--	--	---	--

2.2 Учебно-методическая карта учебной дисциплины

№ недели	Лекции (наименование тем)	Часы	Практические (семинарские) занятия	Часы	Лабораторные занятия	Часы	Самостоятельная Форма контроля знаний	Баллы (max)	
Модуль 1									
1	Тема 1. Введение в язык C++	2	П. з. № 1. Среда программирования Visual Studio. Программирование линейных алгоритмов	2		4	ЗИЗ	4	
2	Тема 2. Управляющие инструкции	2			Л. р. № 1. Программирование ветвящихся алгоритмов	2	4	ЗИЗ	3
3	Тема 3. Компиляция и запуск программы	2	П. з. № 2. Программирование циклических алгоритмов	2			5	ЗИЗ	4
4	Тема 4. Массивы. Строки. Перечисления. Структуры.	2			Л. р. № 2. Программирование с использованием одномерных массивов	2	5	ЗИЗ	3
5	Тема 4. Массивы. Строки. Перечисления. Структуры.	2	П. з. № 3. Указатели. Программирование с использованием динамических двумерных массивов	2			5	ЗИЗ	4
6	Тема 5. Ссылки и указатели	2			Л. р. № 3. Программирование с использованием строк	2	5	ЗИЗ	4
7	Тема 6. Память программы	2	П. з. № 4. Программирование с использованием структур.	2			5	ЗИЗ	4
8	Тема 7. Функции	2			Л. р. № 4. Программирование с использованием функций	2	5	ЗИЗ ПКУ	4 30
9	Тема 8. Многомодульные программы	2	П. з. № 5. Программирование с использованием функций	2			5	ЗИЗ	4
10	Тема 9. Принципы объектно-ориентированного программирования	2			Л. р. № 5. Создание многомодульных программ	2	5	ЗИЗ	4

11	Тема 10. Классы и объекты	2	П. з. № 6. Создание классов и объектов	2			ЗИЗ	4	
12	Тема 11. Специальные методы класса	2			Л.р. № 6. Конструкторы и деструкторы	2	5	ЗИЗ	4
13	Тема 12. Статические члены класса. Дружественные функции и классы	2	П. з. № 7. Статические члены класса	2			5	ЗИЗ	4
14	Тема 13. Перегрузка операций	2			Л.р. № 7. Перегрузка операций	2	5	ЗИЗ	3
15	Тема 14. Наследование классов	2	П. з. № 8. Одиночное наследование	2			5	ЗИЗ	3
16	Тема 15. Виртуальные функции и полиморфизм	2			Л.р. № 8. Полиморфизм и виртуальные методы	2	5	ЗИЗ	2
17	Тема 16. Потоки ввода и вывода	2					5	ЗИЗ ПКУ	2 30
17								ПА (зачет)	40
	Итого	34		16		16	78		100

Принятые обозначения:

Текущий контроль –

ЗИЗ – защита индивидуального задания;

ПКУ – промежуточный контроль успеваемости.

ПА - Промежуточная аттестация.

Итоговая оценка определяется как сумма текущего контроля и промежуточной аттестации и соответствует баллам:

Зачет

Оценка	Зачтено	Не зачтено
Баллы	51-100	0-50

3 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При изучении дисциплины используется модульно-рейтинговая система оценки знаний студентов. Применение форм и методов проведения занятий при изучении различных тем курса представлено в таблице.

№ п/п	Форма проведения занятия*	Вид аудиторных занятий**			Всего часов
		Лекции	Практические занятия	Лабораторные занятия	
1	Мультимедиа	Темы 1-16	–		34
2	С использованием ЭВМ	–	Практ. зан. №№ 1-8		16
3	С использованием ЭВМ	–		Лаб.р. №№ 1 – 8	16
	ИТОГО	34	16	16	66

4 ОЦЕНОЧНЫЕ СРЕДСТВА

Используемые оценочные средства по учебной дисциплине представлены в таблице и хранятся на кафедре.

№ п/п	Вид оценочных средств	Количество комплектов
1	Вопросы к зачетному занятию	1
2	Вопросы к зачетным индивидуальным заданиям	17
3	Тесты по основам программирования	1

5 МЕТОДИКА И КРИТЕРИИ ОЦЕНКИ КОМПЕТЕНЦИЙ СТУДЕНТОВ

5.1 Уровни сформированности компетенций

№ п/п	Уровни сформированности компетенции	Содержательное описание уровня*	Результаты обучения**
<i>Компетенция ПК-7. Способен проводить конструкторские и расчетные работы по проектированию гибких производственных систем в машиностроении</i>			
<i>ИПК-7.1. Способен разрабатывать управляющие программы для гибких производственных систем.</i>			
	1. Пороговый уровень	Понимает базовые принципы написания, оформления и отладки программного кода	Знает методы разработки схем алгоритмов, понимает основные принципы создания прикладных программ.
	2. Продвинутый уровень	Понимает основные принципы написания, оформления и отладки программного кода с помощью принципов структурного программирования и умеет их применять, изредка пользуясь консультацией преподавателя.	Способен создать, отлаживать и отлаживать программный код, пользуясь консультацией преподавателя.
	3. Высокий уровень	Понимает основные принципы написания, оформления и отладки программного кода с помощью принципов структурного программирования и умеет их применять самостоятельно.	Способен самостоятельно создавать и отлаживать программу, отвечающую требованию задания.
<i>ИПК-7.2. Способен работать с персональным компьютером и его периферийными устройствами.</i>			
	1. Пороговый уровень	Понимает базовые принципы работы с персональным компьютером и периферийными устройствами	Знает приемы работы с персональным компьютером и периферийными устройствами.
	2. Продвинутый уровень	Понимает основные принципы взаимодействия персонального компьютера с периферийными устройствами, и умеет их применять при решении	Способен организовать взаимодействие персонального компьютера с периферийными

		профессиональных задач, изредка пользуясь консультацией преподавателя.	устройствами, пользуясь консультацией преподавателя.
	3. Высокий уровень	Понимает основные принципы взаимодействия персонального компьютера с периферийными устройствами, и умеет их самостоятельно применять при решении профессиональных задач.	Способен самостоятельно организовать взаимодействие персонального компьютера с периферийными устройствами, отвечающее требованию задания.
<i>ИПК-7.3 Способен использовать специализированные программные продукты для эмуляции и отладки процесса работы гибких производственных систем.</i>			
	1. Пороговый уровень	Знает назначение и функциональные возможности специализированных программных продуктов для эмуляции и отладки процесса работы гибких производственных систем.	Владеет базовыми навыками работы со специализированными программными продуктами для эмуляции и отладки процесса работы гибких производственных систем.
	2. Продвинутый уровень	Знает назначение и возможности специализированных программных продуктов и может их использовать при решении профессиональных задач, изредка пользуясь консультацией преподавателя.	Умеет применять специализированные программные продукты при решении профессиональных задач, изредка пользуясь консультацией преподавателя.
	3. Высокий уровень	Понимает назначение и возможности специализированных программных продуктов и может самостоятельно их использовать при решении профессиональных задач.	Умеет самостоятельно применять специализированные программные продукты при решении профессиональных задач.

5.2 Методика оценки знаний, умений и навыков студентов

Результаты обучения	Оценочные средства*
<i>Компетенция ПК-7. Способен проводить конструкторские и расчетные работы по проектированию гибких производственных систем в машиностроении</i>	
1 Пороговый уровень	Вопросы к зачётному занятию.

Знает методы разработки программных систем, понимает основные принципы создания прикладных программ.	Вопросы к зачетным индивидуальным заданиям
2 Продвинутый уровень Способен создать, отлаживать и отлаживать программный код, пользуясь консультацией преподавателя.	Вопросы к зачётному занятию. Вопросы к зачетным индивидуальным заданиям
3 Высокий уровень. Способен самостоятельно создавать и отлаживать программы, отвечающие требованию задания.	Вопросы к зачётному занятию. Вопросы к зачетным индивидуальным заданиям

5.3 Критерии оценки лабораторных и практических работ

Каждая лабораторная и практическая работа оценивается от 2 до 4 баллов (см. таблицу п. 2.2, столбец «Баллы»). В оценку включены следующие элементы: выполнение (2 бала), оформление и защита лабораторной работы (3–4 бала). При отсутствии любого элемента работа переходит в разряд задолженности.

5.4 Критерии оценки зачета

На зачете знания, умения и навыки студентов оцениваются следующим образом:

«зачет» – если студент полно освоил учебный материал, владеет понятийным аппаратом, ориентируется в изученном материале, осознанно применяет знания для решения практических задач, грамотно излагает ответ, но содержание, форма ответа имеют отдельные неточности;

«не зачет» – если студент имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал, не может применять знания для решения практических задач.

6 МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Самостоятельная работа студентов (СРС) направлена на закрепление и углубление освоения учебного материала, развитие практических умений. СРС включает следующие виды самостоятельной работы студентов:

Перечень контрольных вопросов и заданий для самостоятельной работы студентов приведен в приложении и хранится на кафедре.

Для СРС рекомендуется использовать источники, приведенные в п. 7.

7 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

7.1 Основная литература

№ п/п	Автор, название, место издания, издательство, год издания учебной литературы	Гриф	Количество экземпляров
1	Гагарина, Л. Г. Технология разработки программного обеспечения : учеб. пособие	Допущено Учебно-методическим	https://znanium.com/catalog/product/1971672

	/ Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Сидорова-Виснадул ; под ред. Л.Г. Гагариной. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2023. — 400 с.	объединением вузов по университетскому политехническому образованию в качестве учебного пособия для студентов высших учебных заведений, обучающихся по направлениям подготовки 09.04.01 и 09.03.03 «Информатика и вычислительная техника»	
--	---	---	--

7.2 Дополнительная литература

№ п/п	Автор, название, место издания, издательство, год издания учебной литературы	Гриф	Количество экземпляров
1	Мякишев, Д.В. Разработка программного обеспечения АСУ ТП на основе объектно-ориентированного подхода : метод. пособие / Д.В. Мякишев. — Москва ; Вологда : Инфра-Инженерия, 2019. - 128 с.	—	https://znanium.com/catalog/product/1048733
2	Дорогов, В. Г. Основы программирования на языке С : учебное пособие / В.Г. Дорогов, Е.Г. Дорогова ; под ред. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2020. — 224 с. — (Высшее образование: Бакалавриат)	Рекомендовано Межрегиональным учебно-методическим советом профессионального образования в качестве учебного пособия для учебных заведений, реализующих программу среднего профессионального образования по укрупненным группам специальностей 09.02.00 «Информатика и вычислительная техника», 10.02.00 «Информационная безопасность»	https://znanium.com/catalog/product/1225391
3	Абрамян, М. Э. Введение в стандартную библиотеку шаблонов C++. Описание, примеры использования, учебные задачи : учебник / М. Э. Абрамян ; Южный федеральный университет. — Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета. 2017. — 178 с. https://znanium.com/catalog/product/1020515	—	https://znanium.com/catalog/product/1020515
4	Корнеев, В. И. Программирование графики на C++. Теория и примеры : учебное пособие / В. И. Корнеев, Л. Г. Гагарина, М. В. Корнеева. — Москва : ФОРУМ : ИНФРА-М, 2019. — 517 с.:	Рекомендовано федеральным государственным автономным образовательным учреждением высшего образования «Национальный исследовательский университет "Московский институт электронной	https://znanium.com/catalog/product/1018909

		техники"» в качестве учебного пособия для студентов, обучающихся по направлению подготовки 09.04.04 «Программная инженерия» и группам направлений 11.03.04 «Электроника и наноэлектроника» и 11.03.02 «Инфокоммуникационные технологии и системы связи»	
--	--	---	--

7.3 Перечень ресурсов сети Интернет по изучаемой дисциплине

1. *Intuit.ru*
2. *Edx.com*

7.4 Перечень наглядных и других пособий, методических рекомендаций по проведению учебных занятий, а также методических материалов к используемым в образовательном процессе техническим средствам

7.4.1 Методические рекомендации

1. Программирование и основы алгоритмизации. Методические рекомендации к лабораторным работам для студентов направления подготовки 15.03.06 Мехатроника и робототехника дневной формы обучения / Сост. Н.Н. Горбатенко. – Могилев: БРУ, 2023 – 29 с.

2. Программирование и основы алгоритмизации. Методические рекомендации к практическим занятиям для студентов направления подготовки 15.03.06 Мехатроника и робототехника дневной формы обучения / Сост. Н.Н. Горбатенко. – Могилев: БРУ, 2023 – 8 с.

7.4.3 Перечень программного обеспечения, используемого в образовательном процессе

Используется свободно распространяемое программное обеспечение.

1. *MS Visual Studio 2019*

8 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Материально-техническое обеспечение дисциплины содержится в паспорте лаборатории «519/2», рег. номер ПУЛ-4 519/2 - 23.

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

по учебной дисциплине Программирование и основы алгоритмизации

направление подготовки 15.03.06 «Мехатроника и робототехника»

направленность (профиль) Робототехника и робототехнические системы:
разработка и применение

на 2024/2025 учебный год

№№ пп	Дополнения и изменения	Основание
1	Дополнений и изменений нет	

Рабочая программа пересмотрена и одобрена на заседании кафедры
«Программное обеспечение информационных технологий»
(название кафедры-разработчика программы)

(протокол № 8 от 29 02 2024)

Заведующий кафедрой

канд. техн. наук., доцент
(ученая степень, ученое звание)



В.В. Кутузов

УТВЕРЖДАЮ

Декан машиностроительного факультета
(название факультета, выпускающего по данной специальности)

канд. техн. наук., доцент
(ученая степень, ученое звание)



Д.М. Свирепа

16. 04 2024

СОГЛАСОВАНО:

Зав. кафедрой «Технология
машиностроения»
(название выпускающей кафедры данной специальности)



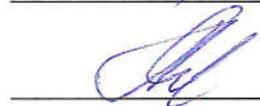
В.М. Шеменков

Ведущий библиотекарь



Р.Н. Киселева

Начальник учебно-методического
отдела



О.Е. Печковская

16. 04 2024