

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования  
«Казанский (Приволжский) федеральный университет»  
Елабужский институт (филиал) КФУ



**УТВЕРЖДАЮ**

Заместитель директора по  
образовательной деятельности

С.Ю. Бахвалов

«19» мая 2025 г.

МП

**Программа дисциплины (модуля)**  
*Программирование*

Направление подготовки/специальность: 15.03.06 Мехатроника и робототехника

Направленность (профиль) подготовки (специальности): Физические основы мехатроники и робототехники

Квалификация: бакалавр

Форма обучения: очная

Язык обучения: русский

Год начала обучения по образовательной программе: - 2025

Программу дисциплины разработал(а)(и) доцент, к.н. Усманов И.Т. (Кафедра математики и прикладной информатики),

### 1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО

Обучающийся, освоивший дисциплину (модуль), должен обладать следующими компетенциями:

Шифр компетенции	Расшифровка приобретаемой компетенции
ОПК-14	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения
ОПК-14.1	Знать технологии разработки алгоритмов и компьютерных программ, пригодных для практического применения
ОПК-14.2	Уметь разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения
ОПК-14.3	Владеть способностью разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения
ПК-2	Способен разрабатывать, отлаживать, внедрять и сопровождать программное обеспечение мехатронных и робототехнических систем
ПК-2.1	Знать способы разработки, отладки и сопровождения программного обеспечения для мехатронных и робототехнических систем
ПК-2.2	Уметь разрабатывать, отлаживать и сопровождать программное обеспечение для мехатронных и робототехнических систем
ПК-2.2	Владеть навыками разработки, отладки и сопровождения программного обеспечения для мехатронных и робототехнических систем

Обучающийся, освоивший дисциплину (модуль):

Должен знать:

технологии разработки алгоритмов и компьютерных программ, пригодных для практического применения в будущей профессиональной деятельности, языки программирования и работы с базами данных, операционные системы и оболочки;

типовые способы разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач.

Должен уметь:

разрабатывать под руководством наставника алгоритмы и компьютерные программы, пригодные для практического применения в будущей профессиональной деятельности, применять языки программирования и современные программные среды разработки информационных систем и технологий;

разрабатывать под руководством наставника программное обеспечение для мехатронных и робототехнических систем при решении конкретных практических задач.

Должен владеть:

способностью разрабатывать под руководством наставника алгоритмы и компьютерные программы, пригодные для практического применения в будущей профессиональной деятельности;

навыками разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач.

### 2. Место дисциплины (модуля) в структуре ОПОП ВО

Данная дисциплина (модуль) включена в Блок 1 "Дисциплины (модули)" Б1.О.05.02 основной профессиональной образовательной программы 15.03.06 «Мехатроника и робототехника» (Физические основы мехатроники и робототехники)" и относится к обязательной части.

Осваивается на 2 курсе в 3, 4 семестрах.

**3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся**

Общая трудоемкость дисциплины составляет 12 зачетных(ые) единиц(ы) на 432 часа(ов).

Контактная работа - 144 часа(ов), в том числе лекции - 60 часа(ов), практические занятия - 0 часа(ов), лабораторные работы - 84 часа(ов), контроль самостоятельной работы - 0 часа(ов).

Самостоятельная работа - 252 часа(ов).

Контроль (зачёт / экзамен) - 36 часа(ов).

Форма промежуточного контроля дисциплины: зачет в 3 семестре; экзамен в 4 семестре.

**4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий**

**4.1 Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)**

N	Разделы дисциплины / модуля	Семестр	Виды и часы контактной работы, их трудоемкость (в часах)			Самостоятельная работа
			Лекции	Практические занятия	Лабораторные работы	
1.	Тема 1. Введение в MicrosoftVisualStudio	3	12	0	18	66
2.	Тема 2. Управляющие операторы языка. Описание и обработка базовых структур	3	12	0	18	66
3.	Тема 3. Модульное программирование. Стандартные модули C#	3	12	0	18	66
4.	Тема 4. Методы проектирования программ	4	4	0	6	14
5.	Тема 5. Основы отладки и тестирования программ. Работа с файлами	4	4	0	8	14
6.	Тема 6. Динамические переменные и указатели	4	6	0	8	14
7.	Тема 7. Основы объектно-ориентированного программирования	4	10	0	8	12
	Итого: 432 часа (из них 36 часа контроль)		60	0	84	252

**4.2 Содержание дисциплины (модуля)**

**Тема 1. Введение в MicrosoftVisualStudio**

Линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии WindowsForms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Windows, WindowsMobile, Windows CE, .NET Framework, Xbox, WindowsPhone .NET CompactFramework.

**Тема 2. Управляющие операторы языка. Описание и обработка базовых структур**

В языке программирования C# существуют специальные операторы, которые в зависимости от вычисляемых значений выражений позволяют управлять ходом выполнения программы. Рассматриваются основные понятия структурного программирования, признаки, типы, различные формы описания и этапы разработки алгоритма

**Тема 3. Модульное программирование. Стандартные модули C#**

Организация программы как совокупности небольших независимых блоков, называемых модулями, структура и поведение которых подчиняются определённым правилам. Использование модульного программирования позволяет упростить тестирование программы и обнаружение ошибок. Аппаратно-зависимые подзадачи могут быть строго отделены от других подзадач, что улучшает мобильность создаваемых программ.

Модуль - библиотека, содержащая описания логически связанных данных (процедур, функций, констант, типов, переменных и т.д.) относящихся к определенной области применения. Язык C# имеет ряд стандартных модулей, которые рассматриваются в visualstudio.

#### **Тема 4. Методы проектирования программ**

Методы проектирования алгоритмов и программ очень разнообразны, их можно классифицировать по различным признакам, важнейшими из которых являются: степень

автоматизации проектных работ; принятая методология процесса разработки.

По степени автоматизации проектирования алгоритмов и программ можно выделить:

методы традиционного (неавтоматизированного) проектирования; методы автоматизированного проектирования (CASE - технология и ее элементы).

#### **Тема 5. Основы отладки и тестирования программ. Работа с файлами**

Комплексная отладка и тестирование программного средства. Тестирование программного средства - это процесс выполнения его программ на некотором наборе данных, для которого заранее известен результат. Получение данных о файлах и каталогах. Чтение и запись текстовых файлов. Сериализация и десериализация объектов. Бинарная сериализация и десериализация. Сериализация и десериализация в формат XML.

#### **Тема 6. Динамические переменные и указатели**

Динамическая переменная создается во время выполнения программы, во многих языках программирования у неё нет собственного идентификатора. Работа с динамической переменной ведётся косвенно, через указатель. Создание такой переменной заключается в выделении участка памяти с помощью специальной функции. Эта функция возвращает адрес в памяти, который назначается указателю. Процесс доступа к памяти через указатель называется разыменованием.

#### **Тема 7. Основы объектно-ориентированного программирования**

В соответствии с принципами объектно-ориентированного программирования решение поставленной задачи сводится к разработке модели (объявлению класса) и созданию экземпляров (объектов), представляющих реализацию этой модели. Обсуждаются проблемы, связанные с созданием и последующим уничтожением объектов

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

Самостоятельная работа обучающихся выполняется по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Самостоятельная работа подразделяется на самостоятельную работу на аудиторных занятиях и на внеаудиторную самостоятельную работу. Самостоятельная работа обучающихся включает как полностью самостоятельное освоение отдельных тем (разделов) дисциплины, так и проработку тем (разделов), осваиваемых во время аудиторной работы. Во время самостоятельной работы обучающиеся читают и конспектируют учебную, научную и справочную литературу, выполняют задания, направленные на закрепление знаний и отработку умений и навыков, готовятся к текущему и промежуточному контролю по дисциплине.

Организация самостоятельной работы обучающихся регламентируется нормативными документами, учебно-методической литературой и электронными образовательными ресурсами, включая:

Порядок организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры (утвержденный приказом Министерства науки и высшего образования Российской Федерации от 6 апреля 2021 года № 245)

Порядок организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры (утвержден приказом Министерства образования и науки Российской Федерации от 5 апреля 2017 года №301)

Письмо Министерства образования Российской Федерации №14-55-996бин/15 от 27 ноября 2002 г. "Об активизации самостоятельной работы студентов высших учебных заведений"

Устав федерального государственного автономного образовательного учреждения "Казанский (Приволжский) федеральный университет"

Правила внутреннего распорядка федерального государственного автономного образовательного учреждения высшего профессионального образования "Казанский (Приволжский) федеральный университет"

Локальные нормативные акты Казанского (Приволжского) федерального университета

### **6. Фонд оценочных средств по дисциплине (модулю)**

Фонд оценочных средств по дисциплине (модулю) включает оценочные материалы, направленные на

проверку освоения компетенций, в том числе знаний, умений и навыков. Фонд оценочных средств включает оценочные средства текущего контроля и оценочные средства промежуточной аттестации.

В фонде оценочных средств содержится следующая информация:

- соответствие компетенций планируемым результатам обучения по дисциплине (модулю);
- критерии оценивания сформированности компетенций;
- механизм формирования оценки по дисциплине (модулю);
- описание порядка применения и процедуры оценивания для каждого оценочного средства;
- критерии оценивания для каждого оценочного средства;
- содержание оценочных средств, включая требования, предъявляемые к действиям обучающихся, демонстрируемым результатам, задания различных типов.

Фонд оценочных средств по дисциплине находится в Приложении 1 к программе дисциплины (модулю).

### 7. Перечень литературы, необходимой для освоения дисциплины (модуля)

Освоение дисциплины (модуля) предполагает изучение учебной литературы. Литература может быть доступна обучающимся в одном из двух вариантов (либо в обоих из них):

- в электронном виде - через электронные библиотечные системы на основании заключенных КФУ договоров с правообладателями;
- в печатном виде - в Научной библиотеке Елабужского института КФУ. Обучающиеся получают учебную литературу на абонементе по читательским билетам в соответствии с правилами пользования Научной библиотекой.

Электронные издания доступны дистанционно из любой точки при введении обучающимся своего логина и пароля от личного кабинета в системе "Электронный университет". При использовании печатных изданий библиотечный фонд должен быть укомплектован ими из расчета не менее 0,25 экземпляра на каждого обучающегося из числа лиц, одновременно осваивающих данную дисциплину

Перечень литературы, необходимой для освоения дисциплины (модуля), находится в Приложении 2 к рабочей программе дисциплины. Он подлежит обновлению при изменении условий договоров КФУ с правообладателями электронных изданий и при изменении комплектования фондов Научной библиотеки Елабужского института КФУ.

### 8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

C# Tutorial - <https://www.tutorialspoint.com/csharp/index.htm>

Основы программирования на C# - <https://www.intuit.ru/studies/courses/2247/18/info>

Проектирование на C# - <http://academy.udmr.ru/kontur/c>

### 9. Методические указания для обучающихся по освоению дисциплины (модуля)

Вид работ	Методические рекомендации
лекции	При изучении дисциплины сначала необходимо по каждой теме лекции прочитать рекомендованную литературу и составить краткий конспект основных положений, терминов, сведений, требующих запоминания и являющихся основополагающими в этой теме для освоения последующих тем курса. Для расширения знания по дисциплине рекомендуется использовать Интернет-ресурсы; проводить поиски в различных системах и использовать материалы сайтов, рекомендованных преподавателем.
лабораторные работы	При подготовке к лабораторным занятиям необходимо заранее изучить методические рекомендации по его проведению. Обратит внимание на цель работы, на основные вопросы для подготовки к работе, на содержание темы работы. Лабораторное занятие проходит в виде диалога, разбора основных вопросов темы. Также лабораторное занятие может проходить в виде показа презентаций, демонстративного материала (в частности плакатов, слайдов), которые сопровождаются беседой преподавателя со студентами. Студент может сдавать лабораторную работу в виде написания реферата, подготовки слайдов, презентаций и последующей защиты его, либо может написать конспект в тетради, ответив на

Вид работ	Методические рекомендации
	вопросы по заданной теме. Ответы на вопросы можно сопровождать рисунками, схемами и т.д. с привлечением дополнительной литературы, которую следует указать
самостоятельная работа	Обучающийся самостоятельно определяет режим своей самостоятельной работы и меру труда, затрачиваемого на овладение знаниями и умениями по дисциплине, выполняет внеаудиторную работу по индивидуальному плану, в зависимости от собственной подготовки, бюджета времени и других условий. Ежедневно обучающийся должен уделять выполнению самостоятельной работы в среднем не менее 3 часов. При выполнении самостоятельной работы обучающийся имеет право обращаться к преподавателю за консультацией с целью уточнения задания, формы контроля выполненного задания.
зачет	Для контроля усвоения данной дисциплины предусмотрен зачет. При ответе на зачете необходимо продумать и четко изложить материал; дать определение основных понятий; явлений. Ответ следует иллюстрировать схемами, рисунками и графиками.
экзамен	Для контроля усвоения данной дисциплины предусмотрен экзамен, на котором студентам необходимо ответить на вопросы экзаменационных билетов. При ответе на экзамене необходимо: продумать и четко изложить материал; дать определение основных понятий; дать краткое описание явлений; привести примеры. Ответ следует иллюстрировать схемами, рисунками и графиками.

#### **10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)**

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем, представлен в Приложении 3 к рабочей программе дисциплины (модуля).

#### **11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)**

Материально-техническое обеспечение образовательного процесса по дисциплине (модулю) включает в себя следующие компоненты:

Учебная аудитория для проведения учебных занятий лекционного типа, семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации № 61

Комплект мебели для преподавателя – 1 шт., посадочные места для обучающихся – 30 шт., одноместные столы – 12 шт., компьютерные столы – 18 шт., компьютеры – 19 шт., интерактивная панель – 1 шт., меловая доска настенная – 1 шт., выход в интернет, внутривузовская компьютерная сеть, доступ в электронную информационно-образовательную среду.

##### **Помещение для самостоятельной работы № 10**

Посадочные места для пользователей – 28 шт., металлические двусторонние стеллажи для книг – 11 шт., книжный шкаф открытый – 5 шт., проектор – 1 шт., ноутбуки для пользователей – 11 шт., шкаф каталожный – 8 шт., шкаф для одежды – 1 шт., ксерокс – 1 шт., рабочий стол библиотекаря – 1 шт., компьютер библиотекаря – 1 шт., вешалка для одежды – 1 шт., жалюзи рулонные «Омега» с фотопечатью – 4 шт., стенд настенный (бронированное стекло) – 4 шт., шкаф-витрина встроенный в арку – 2 шт., шкаф-витрина стеклянный – 2 шт., стеллаж трубчатый с деревянными полками – 2 шт., рабочий стол для инвалидов и лиц с ОВЗ – 2 шт., стол СИ-1 рабочий для инвалидов-колясочников – 1 шт., компьютер – 2 шт., наушники – 2 шт., устройство «Говорящая книга» (тифлоплеер) – 2 шт., видеоувеличитель – 2 шт., радиокласс – 1 шт., портативный тактильный дисплей - 1 шт., сканирующая читающая машина - 1 шт., сканер – 1 шт., веб-камера – 1 шт., выход в интернет, внутривузовская компьютерная сеть, доступ в электронную информационно-образовательную среду.

#### **12. Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья**

При необходимости в образовательном процессе применяются следующие методы и технологии, облегчающие восприятие информации обучающимися инвалидами и лицами с ограниченными возможностями здоровья:

- создание текстовой версии любого нетекстового контента для его возможного преобразования в альтернативные формы, удобные для различных пользователей;

- создание контента, который можно представить в различных видах без потери данных или структуры, предусмотреть возможность масштабирования текста и изображений без потери качества, предусмотреть доступность управления контентом с клавиатуры;

- создание возможностей для обучающихся воспринимать одну и ту же информацию из разных источников - например, так, чтобы лица с нарушениями слуха получали информацию визуально, с нарушениями зрения - аудиально;

- применение программных средств, обеспечивающих возможность освоения навыков и умений, формируемых дисциплиной, за счёт альтернативных способов, в том числе виртуальных лабораторий и симуляционных технологий;

- применение дистанционных образовательных технологий для передачи информации, организации различных форм интерактивной контактной работы обучающегося с преподавателем, в том числе вебинаров, которые могут быть использованы для проведения виртуальных лекций с возможностью взаимодействия всех участников дистанционного обучения, проведения семинаров, выступления с докладами и защиты выполненных работ, проведения тренингов, организации коллективной работы;

- применение дистанционных образовательных технологий для организации форм текущего и промежуточного контроля;

- увеличение продолжительности сдачи обучающимся инвалидом или лицом с ограниченными возможностями здоровья форм промежуточной аттестации по отношению к установленной продолжительности их сдачи:

- продолжительности сдачи зачёта или экзамена, проводимого в письменной форме, - не более чем на 90 минут;

- продолжительности подготовки обучающегося к ответу на зачёте или экзамене, проводимом в устной форме, - не более чем на 20 минут;

- продолжительности выступления обучающегося при защите курсовой работы - не более чем на 15 минут.

Программа составлена в соответствии с требованиями ФГОС ВО и учебным планом по направлению 15.03.06 «Мехатроника и робототехника» и профилю подготовки "Физические основы мехатроники и робототехники".

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего образования  
"Казанский (Приволжский) федеральный университет"  
Елабужский институт (филиал) КФУ

**Фонд оценочных средств по дисциплине**  
*Программирование*

Направление подготовки: 15.03.06 Мехатроника и робототехника

Профиль подготовки: Физические основы мехатроники и робототехники

Квалификация выпускника: бакалавр

Форма обучения: очная

Язык обучения: русский

Год начала обучения по образовательной программе: 2025

## СОДЕРЖАНИЕ

1. Соответствие компетенций планируемым результатам обучения по дисциплине (модулю)
2. Критерии оценивания сформированности компетенций
3. Распределение оценок за формы текущего контроля и промежуточную аттестацию
4. Оценочные средства, порядок их применения и критерии оценивания
  - 4.1. Оценочные средства текущего контроля
    - 4.1.1. Устный опрос
      - 4.1.1.1. Порядок проведения.
      - 4.1.1.2 Критерии оценивания
      - 4.1.1.3. Содержание оценочного средства
    - 4.1.2. Тестирование
      - 4.1.2.1. Порядок проведения.
      - 4.1.2.2 Критерии оценивания
      - 4.1.2.3. Содержание оценочного средства
  - 4.2. Оценочные средства промежуточной аттестации
    - 4.2.1. Устный или письменный ответ на вопрос
      - 4.2.1.1. Зачет
        - 4.2.1.1.1. Порядок проведения и процедура оценивания
        - 4.2.1.1.2. Критерии оценивания
        - 4.2.1.1.3. Оценочные средства
      - 4.2.1.2. Экзамен
        - 4.2.1.2.1. Порядок проведения и процедура оценивания
        - 4.2.1.2.2. Критерии оценивания
        - 4.2.1.2.3. Оценочные средства

1. Соответствие компетенций планируемым результатам обучения по дисциплине (модулю)

Код и наименование компетенции	Индикаторы достижения компетенций для данной дисциплины	Оценочные средства текущего контроля и промежуточной аттестации
<p>ОПК-14. Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</p>	<p>Знать технологии разработки алгоритмов и компьютерных программ, пригодных для практического применения в будущей профессиональной деятельности, языки программирования и работы с базами данных, операционные системы и оболочки.</p> <p>Уметь разрабатывать под руководством наставника алгоритмы и компьютерные программы, пригодные для практического применения в будущей профессиональной деятельности, применять языки программирования и современные программные среды разработки информационных систем и технологий.</p> <p>Владеть способностью разрабатывать под руководством наставника алгоритмы и компьютерные программы, пригодные для практического применения в будущей профессиональной деятельности</p>	<p><b>Текущий контроль:</b> Устный опрос по темам Тема 1. Введение в Microsoft Visual Studio Тема 2. Управляющие операторы языка. Описание и обработка базовых структур Тема 3. Модульное программирование. Стандартные модули C# Тема 4. Методы проектирования программ Тема 5. Основы отладки и тестирования программ. Работа с файлами Тема 6. Динамические переменные и указатели Тема 7. Основы объектно-ориентированного программирования</p> <p>Тестирование по темам: Тема 1. Введение в Microsoft Visual Studio Тема 2. Управляющие операторы языка. Описание и обработка базовых структур Тема 3. Модульное программирование. Стандартные модули C# Тема 4. Методы проектирования программ Тема 5. Основы отладки и тестирования программ. Работа с файлами Тема 6. Динамические переменные и указатели Тема 7. Основы объектно-ориентированного программирования</p> <p><b>Промежуточная аттестация:</b> <i>Зачет</i></p>
<p>ПК-2 Способен разрабатывать, отлаживать, внедрять и сопровождать программное обеспечение мехатронных и робототехнических систем</p>	<p>Знать типовые способы разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач</p> <p>Уметь разрабатывать под руководством наставника программное обеспечение для мехатронных и робототехнических систем при решении конкретных практических задач</p> <p>Владеть навыками разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач</p>	<p><b>Текущий контроль:</b> Устный опрос по темам Тема 1. Введение в Microsoft Visual Studio Тема 2. Управляющие операторы языка. Описание и обработка базовых структур Тема 3. Модульное программирование. Стандартные модули C# Тема 4. Методы проектирования программ Тема 5. Основы отладки и тестирования программ. Работа с файлами Тема 6. Динамические переменные и указатели Тема 7. Основы объектно-ориентированного программирования</p> <p>Тестирование по темам: Тема 1. Введение в Microsoft Visual Studio Тема 2. Управляющие операторы языка. Описание и обработка базовых структур Тема 3. Модульное программирование. Стандартные модули C# Тема 4. Методы проектирования программ Тема 5. Основы отладки и тестирования программ. Работа с файлами Тема 6. Динамические переменные и</p>

		<p>указатели Тема 7. Основы объектно-ориентированного программирования</p> <p><b>Промежуточная аттестация:</b> Экзамен</p>
--	--	--

## 2. Критерии оценивания сформированности компетенций

Компетенция	Зачтено			Не зачтено
	Высокий уровень (отлично) (86-100 баллов)	Средний уровень (хорошо) (71-85 баллов)	Низкий уровень (удовлетворительно) (56-70 баллов)	Ниже порогового уровня (неудовлетворительно) (0-55 баллов)
ОПК-14	Знает технологии разработки алгоритмов и компьютерных программ, пригодных для практического применения в будущей профессиональной деятельности, языки программирования и работы с базами данных, операционные системы и оболочки	Знает основные технологии разработки алгоритмов и компьютерных программ, пригодных для практического применения в будущей профессиональной деятельности, языки программирования и работы с базами данных, операционные системы и оболочки. Допускает незначительные ошибки при ответе на вопрос или решении поставленной задачи	Знает отдельные технологии разработки алгоритмов и компьютерных программ, пригодных для практического применения в будущей профессиональной деятельности, языки программирования и работы с базами данных, операционные системы и оболочки. Допускает типичные ошибки при ответе на вопрос или решении поставленной задачи	Не знает технологии разработки алгоритмов и компьютерных программ, пригодных для практического применения в будущей профессиональной деятельности, языки программирования и работы с базами данных, операционные системы и оболочки
	Умеет разрабатывать под руководством наставника алгоритмы и компьютерные программы, пригодные для практического применения в будущей профессиональной деятельности, применять языки программирования и современные программные среды разработки информационных систем и технологий	Умеет разрабатывать под руководством наставника основные алгоритмы и компьютерные программы, пригодные для практического применения в будущей профессиональной деятельности, применять языки программирования и современные программные среды разработки информационных систем и технологий. Допускает незначительные ошибки при ответе на вопрос или решении поставленной задачи	Умеет разрабатывать под руководством наставника отдельные алгоритмы и компьютерные программы, пригодные для практического применения в будущей профессиональной деятельности, применять языки программирования и современные программные среды разработки информационных систем и технологий. Допускает типичные ошибки при ответе на вопрос или решении поставленной задачи	Не умеет разрабатывать под руководством наставника алгоритмы и компьютерные программы, пригодные для практического применения в будущей профессиональной деятельности, применять языки программирования и современные программные среды разработки информационных систем и технологий
	Владеет способностью разрабатывать под руководством наставника алгоритмы и компьютерные программы, пригодные	Владеет способностью разрабатывать под руководством наставника основные алгоритмы и компьютерные	Владеет способностью разрабатывать под руководством наставника отдельные алгоритмы и компьютерные	Не владеет способностью разрабатывать под руководством наставника алгоритмы и

	для практического применения в будущей профессиональной деятельности.	программы, пригодные для практического применения в будущей профессиональной деятельности. Допускает незначительные ошибки при ответе на вопрос или решении поставленной задачи	программы, пригодные для практического применения в будущей профессиональной деятельности. Допускает типичные ошибки при ответе на вопрос или решении поставленной задачи	компьютерные программы, пригодные для практического применения в будущей профессиональной деятельности.
ПК-2	Знает типовые способы разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач	Знает типовые способы разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач. Допускает незначительные ошибки при ответе на вопрос или решении поставленной задачи	Знает типовые способы разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач. Допускает типичные ошибки при ответе на вопрос или решении поставленной задачи	Не знает типовые способы разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач
	Умеет разрабатывать под руководством наставника программное обеспечение для мехатронных и робототехнических систем при решении конкретных практических задач	разрабатывать под руководством наставника программное обеспечение для мехатронных и робототехнических систем при решении конкретных практических задач. Допускает незначительные ошибки при ответе на вопрос или решении поставленной задачи	Умеет разрабатывать под руководством наставника программное обеспечение для мехатронных и робототехнических систем при решении конкретных практических задач. Допускает типичные ошибки при ответе на вопрос или решении поставленной задачи	Не умеет разрабатывать под руководством наставника программное обеспечение для мехатронных и робототехнических систем при решении конкретных практических задач
	Владеет навыками разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач	Владеет навыками разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач. Допускает незначительные ошибки при ответе на вопрос или решении поставленной задачи	Владеет навыками разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач. Допускает типичные ошибки при ответе на вопрос или решении поставленной задачи	Не владеет навыками разработки программного обеспечения для мехатронных и робототехнических систем при решении конкретных практических задач.

### 3. Распределение оценок за формы текущего контроля и промежуточную аттестацию

Текущий контроль:

3 семестр:

Устный опрос по темам

Тема 1. Введение в Microsoft Visual Studio  
Тема 2. Управляющие операторы языка. Описание и обработка базовых структур  
Тема 3. Модульное программирование. Стандартные модули C#

Максимальное количество баллов по БРС - 30.

Тестирование по темам:

Тема 1. Введение в Microsoft Visual Studio  
Тема 2. Управляющие операторы языка. Описание и обработка базовых структур  
Тема 3. Модульное программирование. Стандартные модули C#

Максимальное количество баллов по БРС - 20.

Итого  $30+20=50$  баллов

**Промежуточная аттестация** – зачет – 50 баллов.

Промежуточная аттестация проводится после завершения изучения дисциплины или ее части в форме, определяемой учебным планом образовательной программы с целью оценить работу обучающегося, степень усвоения теоретических знаний, уровень сформированности компетенций.

Преподаватель, принимающий зачет обеспечивает случайное распределение вариантов заданий между обучающимися с помощью билетов и/или с применением компьютерных технологий; вправе задавать обучающемуся дополнительные вопросы и давать дополнительные задания помимо тех, которые указаны в билете. Зачет проводится по билетам. В каждом билете два оценочных средства: устный или письменный ответ на вопрос и практическое задание.

Устный или письменный ответ – 20 баллов.

Тестирование – 30 баллов.

Итого  $20+30=50$  баллов.

Общее количество баллов по дисциплине за текущий контроль и промежуточную аттестацию:  $50+50=100$  баллов.

Соответствие баллов и оценок:

**Для эзачета:**

56-100 зачтено

0-55 – не зачтено

4 семестр:

Устный опрос по темам:

Тема 4. Методы проектирования программ  
Тема 5. Основы отладки и тестирования программ. Работа с файлами  
Тема 6. Динамические переменные и указатели  
Тема 7. Основы объектно-ориентированного программирования

Максимальное количество баллов по БРС - 30.

Тестирование по темам:

Тема 4. Методы проектирования программ  
Тема 5. Основы отладки и тестирования программ. Работа с файлами  
Тема 6. Динамические переменные и указатели  
Тема 7. Основы объектно-ориентированного программирования

Максимальное количество баллов по БРС - 20.

**Промежуточная аттестация** – экзамен – 50 баллов.

Промежуточная аттестация проводится после завершения изучения дисциплины или ее части в форме, определяемой учебным планом образовательной программы с целью оценить работу обучающегося, степень усвоения теоретических знаний, уровень сформированности компетенций.

Преподаватель, принимающий экзамен обеспечивает случайное распределение вариантов заданий между обучающимися с помощью билетов и/или с применением компьютерных технологий; вправе задавать обучающемуся дополнительные вопросы и давать дополнительные задания помимо тех, которые указаны в билете. Экзамен проводится по билетам. В каждом билете два оценочных средства: устный или письменный ответ на вопрос и практическое задание.

Устный или письменный ответ – 20 баллов.

Тестирование – 30 баллов.

Итого  $20+30=50$  баллов.

Общее количество баллов по дисциплине за текущий контроль и промежуточную аттестацию:  $50+50=100$  баллов.

Соответствие баллов и оценок:

**Для экзамена:**

86-100 – отлично

71-85 – хорошо

56-70 – удовлетворительно

0-55 – неудовлетворительно

#### **4. Оценочные средства, порядок их применения и критерии оценивания**

##### **4.1. Оценочные средства текущего контроля**

###### **4.1.1. Устный опрос по темам:**

3 семестр:

Тема 1. Введение в Microsoft Visual Studio

Visual Studio — это мощное средство разработчика, которое можно использовать для выполнения всего цикла разработки в одном месте. Это комплексная интегрированная среда разработки (IDE), которую можно использовать для записи, редактирования, отладки и сборки кода. Затем разверните приложение. Visual Studio включает компиляторы, средства завершения кода, управление версиями, расширения и многие другие функции для улучшения каждого этапа процесса разработки программного обеспечения.

Тема 2. Управляющие операторы языка. Описание и обработка базовых структур

В теории программирования доказано, что программу для решения задачи любой сложности можно составить из трех структур, называемых следованием (цепочкой), ветвлением и циклом. Этот результат установлен Бойном и Якопини в 1966 г. путем доказательства того, что любую программу можно преобразовать в эквивалентную, состоящую только из этих структур и их комбинаций. Каждая из этих управляющих структур реализована в языке программирования набором соответствующих конструкций.

Тема 3. Модульное программирование. Стандартные модули C#

Под модульностью программ мы понимаем, прежде всего, возможность представления части исходного кода проекта, в виде бинарных независимых модулей. Более того, весь существенный код можно вынести во внешние бинарные модули, тогда как оставшийся код будет выполнять, по сути, только служебные функции.

3 семестр:

Тема 4. Методы проектирования программ

Проектирование алгоритмов и программ - наиболее ответственный этап жизненного цикла программных продуктов, определяющий, насколько создаваемая программа соответствует спецификациям и требованиям со стороны конечных пользователей. Затраты на создание, сопровождение и эксплуатацию программных продуктов, научно-технический уровень разработки, время морального устаревания и многое другое - все это также зависит от проектных решений.

Методы проектирования алгоритмов и программ очень разнообразны, их можно классифицировать по различным признакам, важнейшими из которых являются:

- степень автоматизации проектных работ;
- принятая методология процесса разработки.

По степени автоматизации проектирования алгоритмов и программ можно выделить:

- методы традиционного (неавтоматизированного) проектирования;
- методы автоматизированного проектирования (CASE-технология и ее элементы).

Тема 5. Основы отладки и тестирования программ. Работа с файлами

Необходим строгий контроль правильности каждого этапа решения задачи на компьютере ввиду возможных ошибок в алгоритме, его программной реализации, в исходных данных, в их записи на машинный носитель, в работе аппаратуры, действиях пользователя и т. д. Тема 6. Динамические переменные и указатели

Тема 7. Основы объектно-ориентированного программирования

Объектно ориентированное программирование (сокр. ООП) — методология или стиль программирования на основе описания типов/моделей предметной области и их взаимодействия, представленных порождением из прототипов или как экземпляры классов, которые образуют иерархию наследования.

Идеологически, ООП — подход к программированию как к моделированию информационных объектов, решающий на более высоком абстрактном уровне основную задачу структурного программирования — структурирование информации с точки зрения управляемости. Это позволяет управлять самим процессом моделирования и реализовывать крупные программные проекты.

#### **4.1.1.1. Порядок проведения.**

Устный опрос проводится на занятиях. Обучающиеся выступают с докладами, сообщениями, дополнениями, участвуют в дискуссии, отвечают на вопросы преподавателя. Оценивается уровень домашней подготовки по теме, способность системно и логично излагать материал, анализировать, формулировать собственную позицию, отвечать на дополнительные вопросы.

#### **4.1.1.2 Критерии оценивания**

##### **26-30 баллов ставятся, если обучающимся:**

В ответе качественно раскрыто содержание темы. Ответ хорошо структурирован. Прекрасно освоен понятийный аппарат. Продемонстрирован высокий уровень понимания материала. Превосходное умение формулировать свои мысли, обсуждать дискуссионные положения.

##### **21-25 баллов ставятся, если обучающимся:**

Основные вопросы темы раскрыты. Структура ответа в целом адекватна теме. Хорошо освоен понятийный аппарат. Продемонстрирован хороший уровень понимания материала. Хорошее умение формулировать свои мысли, обсуждать дискуссионные положения.

##### **0-16 баллов ставятся, если обучающимся:**

Тема частично раскрыта. Ответ слабо структурирован. Понятийный аппарат освоен частично. Понимание отдельных положений из материала по теме. Удовлетворительное умение формулировать свои мысли, обсуждать дискуссионные положения.

##### **0-10 баллов ставятся, если обучающимся:**

Тема не раскрыта. Понятийный аппарат освоен неудовлетворительно. Понимание материала фрагментарное или отсутствует. Неумение формулировать свои мысли, обсуждать дискуссионные положения.

#### **4.1.1.3. Содержание оценочного средства**

4 семестр:

1. Опишите систему программирования Microsoft Visual C#.
2. Опишите структуру программ и элементы языка C#.
3. Опишите стандартные функции и выражения языка C#.
4. Опишите процедуры консольного ввода и вывода.
5. Опишите оператор множественного ветвления (выбора).
6. Опишите циклы на языке C#: с предусловием, с постусловием, с параметром.
7. Опишите структурированные типы данных.
8. Опишите двумерные массивы.
9. Опишите алгоритмы поиска в массиве.
10. Опишите алгоритмы сортировки массивов.
11. Как получить список файлов для заданного каталога?
12. Как получить список подкаталогов для заданного каталога?
13. Как выполнить чтение текстового файла в виде единой строки?
14. Как записать текстовый файл в виде единой строки?
15. Как реализовать чтение текстового файла в виде массива строк?
16. Как записать текстовый файл в виде массива строк?
17. Как выполнить сериализацию и десериализацию объекта в бинарный файл?
18. Как осуществить сериализацию и десериализацию объекта в XML-файл?
19. Как управлять сериализацией и десериализацией объекта в XML-файл с использованием атрибутов?
20. Как объявить конструктор класса в языке C#?
21. Как объявить метод класса в языке C#?

#### **4.1.2. Тестирование**

3 семестр:

Тема 1. Введение в Microsoft Visual Studio

Тема 2. Управляющие операторы языка. Описание и обработка базовых структур

Тема 3. Модульное программирование. Стандартные модули C#

4 семестр

Тема 4. Методы проектирования программ

Тема 5. Основы отладки и тестирования программ. Работа с файлами

Тема 6. Динамические переменные и указатели

Тема 7. Основы объектно-ориентированного программирования

#### 4.1.2.1. Порядок проведения.

Тестирование проходит в письменной форме или с использованием компьютерных средств. Обучающийся получает определённое количество тестовых заданий. На выполнение выделяется фиксированное время в зависимости от количества заданий. Оценка выставляется в зависимости от процента правильно выполненных заданий. Тестирование проводится по вариантам.

Ниже приведены примерные задания. Полный банк тестовых заданий хранится на кафедре.

#### 4.1.2.2 Критерии оценивания

17-20 баллов ставится, если обучающийся набрал 86% правильных ответов и более.

14-16 баллов ставится, если обучающийся набрал от 71% до 85 % правильных ответов.

11-13 баллов ставится, если обучающийся набрал от 56% до 70% правильных ответов.

0-10 баллов ставится, если обучающийся набрал 55% правильных ответов и менее

#### 4.1.2.3. Содержание оценочного средства

3 семестр:

Вопрос № 1

Дайте определение языка Си

1. Машинно-ориентированный язык низкого уровня с командами, не всегда соответствующими командам машины, который может обеспечить дополнительные возможности вроде макрокоманд
2. Объектно-ориентированный язык программирования, разработанный компанией Sun Microsystems
3. Компилируемый статически типизированный язык программирования общего назначения

Вопрос № 2

Какие особенности отсутствуют в языке Си

1. Сопрограммы
2. Простая языковая база
3. Области действия имен

Вопрос № 3

Сколько использует программа латинских букв

1. 23
2. 26
3. 31

Вопрос № 4

Что обозначает заголовочный файл

1. Функции ввода/вывода
2. Математическая функция
3. Операции со строками

Вопрос № 5

В каких операциях используется два операнда?

1. Тернарные
2. Унарные
3. Бинарные

Вопрос № 6

Укажите спецификатор "возврат на шаг"

1. \b
2. \v
3. \n

Вопрос № 7

Какой используется оператор, если в программе несколько условий?

1. switch
2. If
3. case

Вопрос № 8

Укажите модификатор, который задает статическую функцию

1. extern
2. void
3. static

Вопрос № 9

С чего начинается программа?

1. #include
2. include irvine32.inc
3. .data

Вопрос № 10

Во что заключаются тело функции?

1. В кавычки
2. В фигурные скобки
3. В квадратные скобки

Вопрос № 11

Укажите директиву нестандартных функций работы с консолью

1. <conio.h>
2. <time.h>
3. <locale.h>

Вопрос № 12

Что означает унарная операция "++"

1. Унарный плюс
2. Сложение
3. Увеличение на 1

Вопрос № 13

Какая используется форматированная строка для вывода информации?

1. scanf
2. float
3. printf

Вопрос № 14

Какое имя должна иметь главная функция программы?

1. void ()
2. main ()
3. void main()

Вопрос № 15

Для чего используется язык Си?

1. Для разработки ОС, драйверов и прикладных программ

2. Для обработки байтового кода
3. Для представления в удобочитаемой форме программ, записанных в машинном коде.

Вопрос № 16

Какого типа язык Си?

1. Интерпретируемого типа
2. Компилируемого типа
3. Интерпретируемого-компилируемого типа

Вопрос № 17

Каким символом должен заканчиваться оператор?

1. Символом "двоеточие" - `:`
2. Символом "точка" - `.`
3. Символом "точка с запятой" - `;`

Вопрос 18

Когда появился язык Си?

1. 1972 год
2. 2001 год
3. 1982 год

Номер вопроса	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Правильный ответ	3	1	2	1	3	1	2	3	1	2	1	3	3	3	1	2	3	1

#### 4.2. Оценочные средства промежуточной аттестации

По дисциплине предусмотрен экзамен во 3 семестре, экзамен в 4 семестре. Они проходят по билетам. В каждом билете один теоретический вопрос и одно практическое задание. Экзамен проводится в устной / письменной и компьютерной форме. Оценивается владение материалом, его системное освоение, способность применять нужные знания, навыки и умения при анализе проблемных ситуаций и решении практических заданий.

##### 4.2.1. Устный или письменный ответ на вопрос

###### 4.2.1.1. Порядок проведения.

Устный или письменный ответ на вопрос направлен на проверку знаний основных разделов по дисциплине «Программирование».

###### 4.2.1.2. Критерии оценивания.

###### 18-20 баллов ставится, если обучающийся:

В ответе качественно раскрыл содержание темы. Ответ хорошо структурирован. Прекрасно освоен понятийный аппарат. Продемонстрирован высокий уровень понимания материала. Превосходное умение формулировать свои мысли, обсуждать дискуссионные положения.

###### 14-17 баллов ставится, если обучающийся:

Основные вопросы темы раскрыл. Структура ответа в целом адекватна теме. Хорошо освоен понятийный аппарат. Продемонстрирован хороший уровень понимания материала. Хорошее умение формулировать свои мысли, обсуждать дискуссионные положения.

###### 11-13 баллов ставится, если обучающийся:

Тему частично раскрыл. Ответ слабо структурирован. Понятийный аппарат освоен частично. Понимание отдельных положений из материала по теме. Удовлетворительное умение формулировать свои мысли, обсуждать дискуссионные положения.

###### 0--10 баллов ставится, если обучающийся:

Тему не раскрыл. Понятийный аппарат освоен неудовлетворительно. Понимание материала фрагментарное или отсутствует. Неумение формулировать свои мысли, обсуждать дискуссионные положения.

###### 4.2.1.3. Оценочные средства.

###### Вопросы для устного или письменного ответа

3 семестр:

1. Система программирования Microsoft Visual C#.
2. Структура программы, элементы языка (алфавит). Понятие типа данных.
3. Операции (арифметические, логические) на типах. Стандартные функции. Выражения.

4. Процедуры консольного ввода и вывода, управление вводом-выводом. Оператор присваивания. Совместимость по присваиванию.
5. Указатели и динамическая память. Динамические переменные.
6. Связанные списки.
7. Очереди. Стеки.
8. Файловые типы. Общие процедуры для работы с файлами. Компонентные (типизированные) файлы.
9. Текстовые файлы. Текст-ориентированные процедуры и функции. Типовые задачи.
10. Прямой и последовательный доступ к компонентам файла. Процедуры и функции, ориентированные на прямой доступ к компонентам файла.
11. Поиск в типизированных файлах. Сортировка файлов (на примере одного из методов).
12. Типизированные файлы. Файлы записей. Типовые алгоритмы обработки.
13. Стандартный модуль Graph. Основные процедуры и функции.
14. Стандартный модуль Graph. Сохранение и выдача изображений на экран.
15. Основные понятия объектно-ориентированного программирования..

4 семестр:

1. Комбинированный тип данных (записи). Оператор присоединения. Записи с вариантами. Программирование типовых алгоритмов обработки записей.
2. Подпрограммы. Основные способы передачи параметров в подпрограмму, их сравнение.
3. Подпрограммы. Область видимости. Локальные и глобальные идентификаторы.
4. Рекурсивные подпрограммы. Область использования.
5. Модуль. Общая структура модуля. Компиляция и подключение модуля.
6. Указатели и динамическая память. Динамические переменные.
7. Связанные списки.
8. Очереди. Стеки.
9. Файловые типы. Общие процедуры для работы с файлами. Компонентные (типизированные) файлы.
10. Текстовые файлы. Текст-ориентированные процедуры и функции. Типовые задачи.
11. Прямой и последовательный доступ к компонентам файла. Процедуры и функции, ориентированные на прямой доступ к компонентам файла.
12. Поиск в типизированных файлах. Сортировка файлов (на примере одного из методов).
13. Типизированные файлы. Файлы записей. Типовые алгоритмы обработки.
14. Стандартный модуль Graph. Основные процедуры и функции.
15. Стандартный модуль Graph. Сохранение и выдача изображений на экран.
16. Основные понятия объектно-ориентированного программирования.

#### 4.2.2. Тестирование

##### 4.2.2.1. Порядок проведения.

Тестирование проходит в письменной форме или с использованием компьютерных средств. Обучающийся получает определённое количество тестовых заданий. На выполнение выделяется фиксированное время в зависимости от количества заданий. Оценка выставляется в зависимости от процента правильно выполненных заданий. Тестирование проводится по вариантам.

Ниже приведены примерные задания. Полный банк тестовых заданий хранится на кафедре.

##### 4.2.2.2. Критерии оценивания.

17-20 баллов ставится, если обучающийся набрал 86% правильных ответов и более.

14-16 баллов ставится, если обучающийся набрал от 71% до 85 % правильных ответов.

11-13 баллов ставится, если обучающийся набрал от 56% до 70% правильных ответов.

0-10 баллов ставится, если обучающийся набрал 55% правильных ответов и менее

##### 4.2.2.3. Оценочные средства.

4 семестр:

Вопрос 1

Выберите верный вариант описания переменной целого типа:

Варианты ответов (выбор одного или несколько вариантов)

1. float b, c;
2. int a;
3. char c, m;
4. int Tu104, П86=23, Yak42;

Вопрос 2

Продолжите фразу: Арифметические выражения, стоящие в правой части оператора присваивания, могут содержать ...

1. целые и вещественные числа, знаки арифметических действий, вызовы стандартных функций, круглые скобки;
- 2) целые и вещественные числа, знаки арифметических действий, вызовы стандартных функций, круглые скобки для изменения порядка действий;
- 3) целые и вещественные числа, знаки арифметических операций, стандартные функции, круглые скобки

#### Вопрос 3

Определите, для обозначения какой арифметической операции используется знак %

1. Вычисление процента
2. целая часть от деления
3. остаток от деления

#### Вопрос 4

Определите назначение стандартной функции, которая вызывается следующим образом: pow(x,y)

1. Остаток от деления x на y
- 2.Целая часть от деления x на y
3. Вычисляет x в степени y

#### Вопрос 5

Вид вызова стандартной функции, извлекающей квадратный корень из вещественного числа x?

1. sqrt(x)
2. sin(x)
3. tan(x)

#### Вопрос 6

Сокращенная запись для следующего оператора присваивания  $x = x + a$ :

1.  $x += a;$
2.  $x+=a$
3.  $x + =a;$

#### Вопрос 7

Определите сокращенную запись для следующего оператора присваивания  $x = x - a$ ;

1.  $x - = a;;$
2.  $X- =a$
3.  $a=x+;$

#### Вопрос 8

Выберите верный формат для вывода значения целого типа (выбор одного или нескольких вариантов):

1. printf("[%d]", 1234);
2. printf("[%d]", i);
3. printf("[%f]", i);
4. printf("[%f]", 123.4);

#### Вопрос 9

Для изменения порядка выполнения операций используются: ...

Варианты ответов

1. Квадратные скобки
2. Круглые скобки
3. Фигурные скобки

#### Вопрос 10

Определите результат работы следующей программы при входных значениях 3 и 5:

```
#include <stdio.h>
```

```
#include <conio.h>
main()
{
int a, b, c;
printf ( "Введите два целых числа \n" );
scanf ( "%d%d", &a, &b );
c = a + b;
printf ( "Результат: %d + %d = %d \n", a, b, c );
getch();
}
```

1. 7
2. 2
3. 8

#### Вопрос 11

Корректно ли выводиться переменная ниже?

```
int num = 0;
printf(num);
```

1. Нет, так как неверно создан вывод
2. Нет, так как неверно создана переменная
3. Да, код верный

#### Вопрос 12

Как подключить стандартную библиотеку «stdio.h»?

1. Только так: #include "stdio.h"
2. Можно двумя способами: #include "stdio.h" и #include <stdio.h>
3. Все варианты верные
4. Только так: #include stdio.h

#### Вопрос 13

Какого типа данных нет в языке Си?

1. Integer
2. Char
3. Float
4. String

#### Вопрос 14

Можно ли создать переменные как показано ниже?

```
int main(void) {
    int a, b, x = 0;
    return 0;
}
```

1. Нет, нельзя. Ничего не будет выведено
2. Нет, нельзя. Будет выведена ошибка
3. Да, можно

#### Вопрос 15

Верно ли прописан код ниже?

```
#include <stdio.h>
int main() {
    float num = 2.5f + 1.8944f;
    printf("%.2f", num);
    return 0;
}
```

Ошибки при выводе и при установке переменной

1. Код верный, всё сработает без ошибок
2. Нет, неверное создание переменной
3. Нет, неверный вывод переменной

#### Вопрос 16

Как указать комментарий?

1. # здесь комментарий
2. // здесь комментарий
3. \*\* здесь комментарий \*\*
4. /\* здесь комментарий \*/

Вопрос 17

С каким значением будет переменная «a»?

```
int a = 2 + 1;
```

1. 1
2. 3
3. Будет ошибка
4. 0

Вопрос 18

Какой результат будет выведен?

```
int main() {
    int a = 10;
    int b = 15;
    int res;

    res = a / b;
    printf("%d", res);

    return 0;
}
```

Будет выведена ошибка

1. Число 0
2. Число 15
3. Число 5
4. Число 10

Номер вопроса	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Правильный ответ	2, 4	3	Зв	1	1	1	2	1, 2	2	3	1	2	4	3	1	2	3	2

**Перечень литературы, необходимой для освоения дисциплины (модуля)**

Направление подготовки: 15.03.06 Мехатроника и робототехника

Профиль подготовки: Физические основы мехатроники и робототехники

Квалификация выпускника: бакалавр

Форма обучения: очная

Язык обучения: русский

Год начала обучения по образовательной программе: 2025

**Основная литература:**

1. Агафонов, Е. Д. Прикладное программирование: учебное пособие / Е. Д. Агафонов, Г. В. Ващенко. - Красноярск: СФУ, 2015. - 112 с. - ISBN 978-5-7638-3165-8. - Текст: электронный. - URL: <https://znanium.com/catalog/product/550046>. - Режим доступа: по подписке.
2. Медведев, М. А. Программирование на СИ#: Учебное пособие / Медведев М.А., Медведев А.Н., - 2-е изд., стер. - Москва: Флинта, Изд-во Урал.ун-та, 2017. - 64 с. ISBN 978-5-9765-3169-7. - Текст: электронный. - URL: <https://znanium.com/catalog/product/948428>. - Режим доступа: по подписке.
3. Федотова, Е. Л. Информатика. Курс лекций : учеб.пособие / Е. Л. Федотова, А. А. Федотов. — Москва : ФОРУМ, ИНФРА-М, 2018. — 480 с. - (Высшее образование). - ISBN 978-5-8199-0448-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/914260>. - Режим доступа: по подписке.
4. Царев, Р.Ю. Информатика и программирование [Электронный ресурс] : учеб.пособие / Р. Ю. Царев, А. Н. Пупков, В. В. Самарин, Е. В. Мыльникова. - Красноярск: Сиб. федер. ун-т, 2014. - 132 с. - ISBN 978-5-7638-3008-8. - Текст: электронный. - URL: <https://znanium.com/catalog/product/506203>. - Режим доступа: по подписке.

**Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем**

Направление подготовки: 15.03.06 Мехатроника и робототехника

Профиль подготовки: Физические основы мехатроники и робототехники

Квалификация выпускника: бакалавр

Форма обучения: очная

Язык обучения: русский

Год начала обучения по образовательной программе: 2025

Освоение дисциплины (модуля) предполагает использование следующего программного обеспечения и информационно-справочных систем:

Программное обеспечение: операционная система Windows, Microsoft Office, PyCharm,

Kaspersky Free для Windows, деловая игра: корпорация плюс. Project Expert 7, 1С: Предприятие 8.3 Учебная версия

Электронная библиотечная система «ZNANIUM.COM»

Электронная библиотечная система Издательства «Лань»

Электронная библиотечная система «Консультант студента»