

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Максимов Алексей Борисович

Должность: директор департамента по образовательной политике

Дата подписания: 12.10.2026 17:00:48

Уникальный идентификатор:

8db180d1a3f02ac9e60521a5672742735c18b1d6

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение
высшего образования**

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ

Декан факультета

«Информационных технологий»



/ Д.Г. Демидов /

« 26 » февраля 2026 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Языки высокого уровня**

Направление подготовки

09.03.01 Информатика и вычислительная техника

Профиль подготовки (образовательная программа)

«Программирование электронных устройств и систем»

Год начала обучения:

2026

Квалификация (степень) выпускника

Бакалавр

Форма обучения

Очная

Москва – 2026

Рабочая программа составлена на основе ФГОС ВО и учебного плана Московского политехнического университета по направлению (специальности) 09.03.01 Информатики и вычислительная техника, по профилю подготовки Программирование электронных устройств и систем.

Разработчик(и):

к. ф.-м. н., доцент кафедры



/ Т.Т. Идиатуллов /

Согласовано:

Заведующий кафедрой
«СМАРТ-технологии»,
к.т.н., доцент



/ Е.В. Петрунина /

Содержание

1. Цели, задачи и планируемые результаты обучения по дисциплине
2. Место дисциплины в структуре образовательной программы
3. Структура и содержание дисциплины
 - 3.1 Виды учебной работы и трудоемкость
 - 3.2 Тематический план изучения дисциплины (по формам обучения)
 - 3.3 Содержание дисциплины
 - 3.4 Тематика семинарских/практических и лабораторных занятий
 - 3.5 Тематика вопросов для самостоятельного изучения
4. Учебно-методическое и информационное обеспечение
 - 4.1 Нормативные документы и ГОСТы
 - 4.2. Основная литература
 - 4.3 Дополнительная литература
 - 4.4 Электронные образовательные ресурсы
 - 4.5 Лицензионное и свободно распространяемое программное обеспечение
 - 4.6 Современные профессиональные базы данных и информационные справочные системы
- 5 Материально-техническое обеспечение
- 6 Методические рекомендации
 - 6.1 Методические рекомендации для преподавателя по организации обучения
 - 6.2 Методические указания для обучающихся по освоению дисциплины
 - 6.3 Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья
- 7 Фонд оценочных средств
 - 7.1 Методы контроля и оценивания результатов обучения
 - 7.2 Шкала и критерии оценивания результатов обучения
- Приложение 1
- Приложение 2

1 Цели, задачи и планируемые результаты обучения по дисциплине

К основным целям освоения дисциплины «Языки высокого уровня» следует отнести:

- формирование у обучающихся системного представления о конструкциях языков высокого уровня и парадигмах программирования: процедурной и объектно-ориентированной;
- ознакомление обучающихся с методами разработки, отладки и тестирования программ на языках высокого уровня, включая работу с динамической памятью, объектами, шаблонами и стандартной библиотекой;
- формирование навыков разработки программного обеспечения на языках высокого уровня для решения прикладных задач, в том числе для устройств и информационных систем умных пространств.

К основным задачам освоения дисциплины «Языки высокого уровня» следует отнести:

- изучение основных конструкций языка высокого уровня: типов данных, операторов, ветвлений, циклов;
- изучение функций, способов передачи параметров, рекурсии;
- изучение работы с массивами, указателями и динамической памятью;
- изучение структур данных и работы с файлами;
- изучение основ объектно-ориентированного программирования: классов, инкапсуляции, наследования, полиморфизма;
- изучение обобщенного программирования (шаблонов) и стандартной библиотеки шаблонов;
- изучение механизмов обработки исключений;
- приобретение навыков разработки, отладки и тестирования программ на языке высокого уровня для устройства умного пространства.

Обучение по дисциплине «Языки высокого уровня» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	Индикаторы достижения компетенции
ОПК-7. Способен участвовать в настройке и	ИОПК-7.1. Знает принципы взаимодействия программ на языках высокого уровня с

наладке программно-аппаратных комплексов	аппаратной частью вычислительной системы и периферийными устройствами ИОПК-7.2. Умеет разрабатывать программы на языках высокого уровня для настройки и наладки программно-аппаратных комплексов ИОПК-7.3. Имеет навыки отладки программ на языках высокого уровня при работе с аппаратными компонентами устройства
ОПК-8. Способен разрабатывать алгоритмы и реализовывать их программно с использованием языков программирования и современных инструментальных средств	ИОПК-8.1. Знает конструкции языков высокого уровня и парадигмы программирования: процедурную и объектно-ориентированную ИОПК-8.2. Умеет разрабатывать и реализовывать алгоритмы на языках высокого уровня с использованием современных инструментальных средств ИОПК-8.3. Имеет навыки разработки, отладки и тестирования программ на языках высокого уровня

2 Место дисциплины в структуре образовательной программы

Дисциплина «Языки высокого уровня» относится к числу общепрофессиональных и профессиональных учебных дисциплин основной образовательной программы бакалавриата по направлению подготовки 09.03.01 «Информатика и вычислительная техника», профиль «Программирование электронных устройств и систем».

Изучение дисциплины опирается на знания, умения и навыки, приобретенные в предшествующих дисциплинах: «Информатика», «Языки программирования», «Дискретная математика».

Дисциплина обеспечивает изучение дисциплин «Технологии программирования инженерных систем», «Базы данных», «Программирование микроконтроллеров», а также выполнение курсовых работ и выпускной квалификационной работы.

3 Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 8 зачетных единиц, т.е. 288 академических часов, из них аудиторных — 90 часов, самостоятельная работа обучающихся — 198 часов. Дисциплина изучается в течение двух семестров. Форма промежуточной аттестации — дифференцированный зачет в 5 семестре и экзамен в 6 семестре.

3.1 Виды учебной работы и трудоемкость

(по очной форме обучения)

№ п/п	Вид учебной работы	Количество часов	Семестр 5	Семестр 6
1	Аудиторные занятия	90	42	48
	В том числе:			
1.1	Лекции	36	18	18
1.2	Семинарские/практические занятия	-	-	-
1.3	Лабораторные занятия	54	24	30
2	Самостоятельная работа	198	90	108
3	Промежуточная аттестация	диф. зачет / экзамен	дифференцированный зачет	экзамен
	Итого:	288	132	156

3.2 Тематический план изучения дисциплины

(по очной форме обучения)

3.2.1 Тематический план (5 семестр)

№ п/п	Разделы/темы дисциплины	Всего, час.	Лекции	Лабораторные занятия	Самостояте льная работа
1	Тема 1. Введение в языки высокого уровня. Основные конструкции: типы данных, операторы, ветвления, циклы	32	4	6	22
2	Тема 2. Функции. Передача параметров. Рекурсия	32	4	6	22

3	Тема 3. Массивы и указатели. Работа с динамической памятью	34	5	6	23
4	Тема 4. Структуры и объединения. Работа с файлами	34	5	6	23
	Итого	132	18	24	90

3.2.2 Тематический план (6 семестр)

№ п/п	Разделы/темы дисциплины	Всего, час.	Лекции	Лабораторные занятия	Самостояте льная работа
5	Тема 5. Основы объектно-ориентированного программирования: классы, инкапсуляция, конструкторы и деструкторы	26	3	5	18
6	Тема 6. Наследование и полиморфизм. Виртуальные функции	26	3	5	18
7	Тема 7. Обобщенное программирование: шаблоны и стандартная библиотека шаблонов (STL)	34	4	6	24
8	Тема 8. Обработка исключений. Комплексное проектирование программного проекта для устройства умного пространства	70	8	14	48
	Итого	156	18	30	108

3.3 Содержание дисциплины

3.3.1 Содержание лекционных занятий (36 часов)

Тема 1. Введение в языки высокого уровня. Основные конструкции: типы данных, операторы, ветвления, циклы (4 ч)

Понятие языка программирования высокого уровня, соотношение с языками низкого уровня. Обзор основных языков высокого уровня и областей их применения. Базовые типы данных, операции над ними, приведение типов. Операторы ветвления: условный оператор, оператор множественного выбора. Операторы цикла: цикл с предусловием, с постусловием, с известным числом повторений. Правила оформления программного кода, понятие стиля программирования.

Тема 2. Функции. Передача параметров. Рекурсия (4 ч)

Понятие функции, назначение декомпозиции программы на функции. Объявление и определение функции, прототипы. Способы передачи параметров: по значению, по ссылке, по указателю. Область видимости переменных, время жизни переменных. Понятие рекурсии, примеры рекурсивных алгоритмов, сравнение рекурсивного и итеративного подходов.

Тема 3. Массивы и указатели. Работа с динамической памятью (5 ч)

Одномерные и многомерные массивы: объявление, инициализация, обращение к элементам. Понятие указателя, операции над указателями, связь массивов и указателей. Динамическое выделение и освобождение памяти. Типичные ошибки при работе с указателями и динамической памятью: утечки памяти, разыменование недействительных указателей.

Тема 4. Структуры и объединения. Работа с файлами (5 ч)

Понятие структуры, объявление структурного типа, обращение к полям структуры. Массивы структур, вложенные структуры. Понятие объединения, отличие от структуры. Организация ввода-вывода данных в файлы: текстовые и бинарные файлы, основные операции с файлами. Обработка ошибок при работе с файлами.

Тема 5. Основы объектно-ориентированного программирования: классы, инкапсуляция, конструкторы и деструкторы (3 ч)

Понятие объектно-ориентированного программирования, основные принципы: инкапсуляция, наследование, полиморфизм. Понятие класса как пользовательского типа данных, объекта как экземпляра класса. Модификаторы доступа к членам класса. Конструкторы и деструкторы, их назначение и особенности вызова. Перегрузка конструкторов, конструктор копирования.

Тема 6. Наследование и полиморфизм. Виртуальные функции (3 ч)

Понятие наследования, отношение «является» между классами. Одиночное и множественное наследование, модификаторы наследования. Понятие полиморфизма, статическое и динамическое связывание. Виртуальные функции, механизм их вызова через указатель или ссылку на базовый класс. Абстрактные классы и чисто виртуальные функции.

Тема 7. Обобщенное программирование: шаблоны и стандартная библиотека шаблонов (STL) (4 ч)

Понятие обобщенного программирования, назначение шаблонов функций и классов. Синтаксис объявления шаблонов, инстанцирование шаблонов компилятором. Обзор стандартной библиотеки шаблонов: контейнеры (векторы, списки, множества, словари), итераторы, алгоритмы. Применение стандартных контейнеров и алгоритмов для решения типовых задач обработки данных.

Тема 8. Обработка исключений. Комплексное проектирование программного проекта для устройства умного пространства (8 ч)

Понятие исключительной ситуации, механизм обработки исключений: конструкция try-catch, генерация исключений. Иерархия исключений, разработка пользовательских классов исключений. Особенности использования исключений для повышения надежности программного обеспечения. Этапы разработки комплексного программного проекта: формирование требований, проектирование архитектуры программы, реализация, тестирование, отладка. Особенности разработки программного обеспечения на языке высокого уровня для устройств умного пространства, включая взаимодействие с периферийными устройствами и оптимизацию по ресурсам.

3.3.2 Содержание лабораторных занятий (54 часа)

Лабораторная работа № 1. Разработка программ с использованием базовых операторов и ветвлений (4 ч)

Разработка программ, реализующих обработку данных с использованием условных операторов и оператора множественного выбора. Отладка разработанных программ с использованием отладчика.

Лабораторная работа № 2. Разработка программ с циклическими конструкциями (4 ч)

Разработка программ обработки последовательности данных с использованием различных видов циклов. Оптимизация циклических конструкций по количеству выполняемых операций.

Лабораторная работа № 3. Разработка и отладка функций, реализация рекурсивных алгоритмов (4 ч)

Декомпозиция программы на функции для решения составной задачи. Реализация и отладка рекурсивного алгоритма, сравнение с итеративной реализацией по эффективности.

Лабораторная работа № 4. Работа с массивами и указателями (4 ч)

Разработка программ обработки одномерных и двумерных массивов. Реализация типовых алгоритмов работы с массивами с использованием указателей.

Лабораторная работа № 5. Работа с динамической памятью (4 ч)

Разработка программы, использующей динамическое выделение памяти для структур данных переменного размера. Проверка корректности освобождения выделенной памяти.

Лабораторная работа № 6. Работа со структурами и файлами (4 ч)

Разработка программы обработки набора структур данных. Реализация сохранения и загрузки данных из текстового или бинарного файла.

Лабораторная работа № 7. Разработка классов: инкапсуляция, конструкторы и деструкторы (4 ч)

Разработка класса для представления объекта предметной области с инкапсуляцией данных. Реализация конструкторов, деструктора и методов доступа к полям класса.

Лабораторная работа № 8. Реализация наследования и полиморфизма (4 ч)

Разработка иерархии классов с использованием наследования. Реализация полиморфного поведения объектов производных классов.

Лабораторная работа № 9. Работа с виртуальными функциями и абстрактными классами (4 ч)

Разработка абстрактного базового класса с чисто виртуальными функциями. Реализация производных классов и демонстрация динамического связывания при вызове виртуальных функций.

Лабораторная работа № 10. Разработка обобщенных функций и классов с использованием шаблонов (4 ч)

Разработка шаблонной функции для выполнения типовой операции над данными различных типов. Разработка шаблонного класса-контейнера.

Лабораторная работа № 11. Использование контейнеров и алгоритмов STL (4 ч)

Решение прикладной задачи обработки данных с использованием стандартных контейнеров (вектор, список, словарь) и алгоритмов стандартной библиотеки шаблонов.

Лабораторная работа № 12. Обработка исключений (4 ч)

Разработка программы с обработкой исключительных ситуаций средствами конструкции try-catch. Разработка пользовательского класса исключения для специфической ошибки предметной области.

Лабораторная работа № 13. Комплексный проект: разработка программного модуля для устройства умного пространства (6 ч)

Выполнение комплексного проекта по разработке программного модуля на языке высокого уровня для заданного устройства умного пространства с применением объектно-ориентированного подхода, шаблонов и обработки исключений. Оформление отчета и защита результатов проекта.

3.3.3 Содержание самостоятельной работы обучающихся (198 часов)

Самостоятельная работа обучающихся включает следующие виды деятельности:

- проработка конспектов лекций и изучение разделов дисциплины, вынесенных на самостоятельное изучение, по рекомендованной литературе и электронным образовательным ресурсам;
- изучение документации языка программирования и стандартной библиотеки;
- подготовка к выполнению и защите практических работ, оформление отчетов;
- выполнение расчетно-графических заданий по разработке алгоритмов и программ различной сложности;
- подготовка обзора (реферата) по особенностям современных языков программирования высокого уровня;
- выполнение индивидуального задания по разработке программного обеспечения для устройства умного пространства;
- подготовка к тестированию, к дифференцированному зачету в 5 семестре и к экзамену в 6 семестре.

4 Учебно-методическое и информационное обеспечение

4.1 Нормативные документы и ГОСТы

1. Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации».
2. Приказ Минобрнауки России от 19.09.2017 № 929 (ред. от 08.02.2021) «Об утверждении федерального государственного образовательного стандарта высшего образования — бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника» (Зарегистрировано в Минюсте России 10 октября 2017 г. № 48489).
3. Академический учебный план. Направление подготовки: 09.03.01 Информатика и вычислительная техника. Профиль: Программирование электронных устройств и систем. Форма обучения: очная.
4. Положение о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся федерального государственного автономного образовательного учреждения высшего образования «Московский политехнический университет» (утверждено приказом Московского Политеха от 01.12.2022 № 1375ОД).

4.2. Основная литература

1. Керниган, Б. Язык программирования Си / Б. Керниган, Д. Ритчи. — Санкт-Петербург : Невский Диалект, 2009. — 352 с.
2. Страуструп, Б. Язык программирования C++ / Б. Страуструп. — Москва : Бином, 2011. — 1136 с.
3. Липпман, С. Язык программирования C++. Базовый курс / С. Липпман, Ж. Лажойе, Б. Му. — Москва : Вильямс, 2014. — 1120 с.

4.3 Дополнительная литература

1. Мейерс, С. Эффективный и современный C++ / С. Мейерс. — Санкт-Петербург : Питер, 2016. — 304 с.
2. Прата, С. Язык программирования C++. Лекции и упражнения / С. Прата. — Москва : Вильямс, 2019. — 1120 с.
3. Шилдт, Г. C++: базовый курс / Г. Шилдт. — Москва : Вильямс, 2010. — 624 с.

4.4 Электронные образовательные ресурсы

1. Курс в ЛМС Московского политехнического университета. — <https://lms.mospolytech.ru/>
2. Электронно-библиотечная система «Лань». — <https://e.lanbook.com/>

3. Электронно-библиотечная система «Юрайт». — <https://urait.ru/>
4. Справочная документация по стандартной библиотеке языка C++. — <https://en.cppreference.com/>

4.5 Лицензионное и свободно распространяемое программное обеспечение

1. Операционная система Windows 10 (или выше) — Microsoft Open License.
2. Офисные приложения Microsoft Office 2016 (или выше) — Microsoft Open License.
3. Интегрированная среда разработки (например, Visual Studio Code либо отечественный аналог) — свободно распространяемое ПО.
4. Компилятор языка программирования (например, GCC/G++, либо отечественный аналог) — свободно распространяемое ПО.

4.6 Современные профессиональные базы данных и информационные справочные системы

1. Информационная система «Техэксперт»: нормативно-техническая документация. — <https://cntd.ru/>
2. Электронный фонд правовой и нормативно-технической документации. — <https://docs.cntd.ru/>

5 Материально-техническое обеспечение

Для проведения всех видов занятий необходимо презентационное оборудование (мультимедийный проектор, компьютер, экран) — 1 комплект.

Для проведения практических занятий необходимо наличие компьютерного класса, оборудованного современной вычислительной техникой из расчета одно рабочее место на одного обучающегося с доступом в сеть Интернет.

6 Методические рекомендации

6.1 Методические рекомендации для преподавателя по организации обучения

При подготовке к занятиям следует предварительно проработать материал занятия, предусмотрев его подачу точно в отведенное для этого время занятия. Следует подготовить необходимые материалы — примеры программного кода, наборы тестовых заданий, задания для практических занятий.

Практические занятия целесообразно выстраивать по принципу постепенного усложнения: в 5 семестре — от базовых конструкций языка к

работе с указателями, динамической памятью и файлами; в 6 семестре — от основ объектно-ориентированного программирования к шаблонам, стандартной библиотеке и комплексному проектированию программного обеспечения.

При проведении занятий рекомендуется использовать разбор реальных программных проектов, применяемых при разработке устройств умных пространств, чтобы обучающиеся видели практическую значимость изучаемого материала.

При проверке работ и отчетов следует учитывать не только формальное соответствие требованиям задания, но и качество программного кода, корректность и эффективность разработанных алгоритмов.

6.2 Методические указания для обучающихся по освоению дисциплины

Изучение дисциплины осуществляется в строгом соответствии с целевой установкой в тесной взаимосвязи с учебным планом. Основой теоретической подготовки обучающихся являются лекции.

В процессе самостоятельной работы обучающиеся закрепляют и углубляют знания, полученные во время аудиторных занятий, дорабатывают конспекты лекций, изучают документацию языка программирования и стандартной библиотеки, готовятся к защите практических работ, к дифференцированному зачету в 5 семестре и к экзамену в 6 семестре, а также самостоятельно изучают отдельные темы учебной программы.

Особое внимание при подготовке следует уделять приобретению практических навыков написания и отладки программного кода. Рекомендуется регулярно практиковаться в решении задач различной сложности и анализировать примеры качественного программного кода.

6.3 Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья

При необходимости для обучающихся инвалидов и лиц с ограниченными возможностями здоровья с учетом их индивидуальных потребностей и по личному заявлению обеспечивается: предоставление учебных материалов в доступных форматах, использование при необходимости специализированных технических средств обучения, увеличение продолжительности выполнения практических и контрольных работ, выбор мест прохождения практики и проведения занятий с учетом состояния здоровья обучающегося, а также иные условия, предусмотренные локальными нормативными актами университета.

7 Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

В процессе обучения используются следующие оценочные формы самостоятельной работы обучающихся, оценочные средства текущего контроля успеваемости и промежуточной аттестации:

- подготовка к решению прикладных задач;
- выполнение и защита практических работ, оформление отчетов;
- выполнение расчетно-графических заданий и их защита;
- выполнение индивидуального задания по разработке программного обеспечения для устройства умного пространства;
- подготовка обзора (реферата) по особенностям современных языков программирования;
- тест;
- дифференцированный зачет (5 семестр);
- экзамен (6 семестр).

7.2 Шкала и критерии оценивания результатов обучения

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине (модулю).

ОПК-7. Способен участвовать в настройке и наладке программно-аппаратных комплексов

Показатель	2	3	4	5
знать: принципы взаимодействия программ на языках высокого уровня с аппаратной частью вычислительной системы	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие указанных знаний	Обучающийся демонстрирует неполное соответствие знаний, освещена основная, наиболее важная часть материала, допущена одна значительная ошибка или неточность	Обучающийся демонстрирует частичное соответствие указанных знаний, допускаются 2–3 незначительные ошибки неточности	Обучающийся демонстрирует полное соответствие указанных знаний, свободно оперирует материалом дисциплины
уметь: разрабатывать программы на языках	Обучающийся не умеет или в	Обучающийся демонстрирует	Обучающийся демонстрирует	Обучающийся демонстрирует

высокого уровня для настройки и наладки программно-аппаратных комплексов	недостаточной степени умеет выполнять указанные действия	неполное соответствие указанных умений, испытывает значительные затруднения при переносе умений на новые ситуации	частичное соответствие указанных умений, допускает незначительные ошибки и неточности	полное соответствие указанных умений, уверенно применяет их в нестандартных ситуациях
владеть: навыками отладки программ на языках высокого уровня при работе с аппаратными компонентами устройства	Обучающийся не владеет или в недостаточной степени владеет указанными навыками	Обучающийся владеет указанными навыками, но допускаются значительные ошибки, проявляется недостаточность навыков по ряду показателей	Обучающийся владеет указанными навыками, допускаются незначительные ошибки и затруднения при практических операциях	Обучающийся в полном объеме владеет указанными навыками, свободно оперирует приобретенными знаниями

ОПК-8. Способен разрабатывать алгоритмы и реализовывать их программно с использованием языков программирования и современных инструментальных средств

Показатель	2	3	4	5
знать: конструкции языков высокого уровня, парадигмы процедурного и объектно-ориентированного программирования	Обучающийся демонстрирует полное или недостаточное соответствие указанных знаний	Обучающийся демонстрирует неполное соответствие знаний, освещена основная, наиболее важная часть материала, допущена одна значительная ошибка или неточность	Обучающийся демонстрирует частичное соответствие указанных знаний, допускаются 2–3 незначительные ошибки и неточности	Обучающийся демонстрирует полное соответствие указанных знаний, свободно оперирует материалом дисциплины
уметь: разрабатывать и реализовывать алгоритмы на языках высокого уровня с использованием современных инструментальных средств	Обучающийся не умеет или в недостаточной степени умеет выполнять указанные действия	Обучающийся демонстрирует неполное соответствие указанных умений, испытывает значительные	Обучающийся демонстрирует частичное соответствие указанных умений, допускает незначительные	Обучающийся демонстрирует полное соответствие указанных умений, уверенно применяет их в

		затруднения при переносе умений на новые ситуации	ошибки неточности	и нестандартных ситуациях
владеть: навыками разработки, отладки и тестирования программ на языках высокого уровня	Обучающийся не владеет или в недостаточной степени владеет указанными навыками	Обучающийся владеет указанными навыками, но допускаются значительные ошибки, проявляется недостаточность навыков по ряду показателей	Обучающийся владеет указанными навыками, допускаются незначительные ошибки и затруднения при практических операциях	Обучающийся в полном объеме владеет указанными навыками, свободно оперирует приобретенными знаниями

Шкалы оценивания результатов промежуточной аттестации и их описание:

Форма промежуточной аттестации (5 семестр): дифференцированный зачет.

Промежуточная аттестация обучающихся в форме дифференцированного зачета проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине (модулю) в 5 семестре, при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине (модулю) проводится преподавателем, ведущим занятия по дисциплине (модулю), методом экспертной оценки. По итогам промежуточной аттестации выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

Шкала оценивания	Описание
отлично	Выполнены все виды учебной работы, предусмотренные учебным планом. Обучающийся демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, свободно оперирует приобретенными знаниями, умениями и навыками, применяет их в ситуациях повышенной сложности. Все практические работы выполнены и защищены. Присутствовал более чем на $\frac{3}{4}$ занятий по дисциплине.

хорошо	Выполнены все виды учебной работы, предусмотренные учебным планом. Обучающийся демонстрирует неполное, но правильное соответствие знаний, умений, навыков приведенным в таблицах показателей, либо при этом были допущены 2–3 несущественные ошибки. Присутствовал более чем на $\frac{3}{4}$ занятий по дисциплине.
удовлетворительно	Выполнены все виды учебной работы, предусмотренные учебным планом. Обучающийся демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность. Присутствовал более чем на $\frac{1}{2}$ занятий по дисциплине.
неудовлетворительно	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Обучающийся демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, обучающийся испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

Форма промежуточной аттестации (6 семестр): экзамен.

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине (модулю) в 6 семестре, а также по результатам устного ответа обучающегося на вопросы экзаменационного билета. При выставлении оценки учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине (модулю) проводится преподавателем, ведущим занятия по дисциплине (модулю), методом экспертной оценки.

Оценка	Критерии оценивания
5 (отлично)	Выполнены все виды учебной работы, предусмотренные учебным планом. Обучающийся демонстрирует глубокие и систематизированные знания материала дисциплины, свободно оперирует основными понятиями, уверенно и полно отвечает

	на вопросы экзаменационного билета и дополнительные вопросы.
4 (хорошо)	Выполнены все виды учебной работы, предусмотренные учебным планом. Обучающийся демонстрирует хорошие знания материала дисциплины, достаточно полно отвечает на вопросы экзаменационного билета, но допускает отдельные неточности, устраняемые с помощью уточняющих вопросов.
3 (удовлетворительно)	Выполнены основные виды учебной работы, предусмотренные учебным планом. Обучающийся демонстрирует знание основного материала дисциплины, но отвечает на вопросы экзаменационного билета неполно либо с существенными неточностями, требующими наводящих вопросов.
2 (неудовлетворительно)	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Обучающийся не может ответить на вопросы экзаменационного билета либо демонстрирует существенные пробелы в знаниях материала дисциплины, не устраняемые дополнительными вопросами.

7.3 Оценочные средства

7.3.1 Список вопросов для дифференцированного зачета (5 семестр)

1. Понятие языка программирования высокого уровня, соотношение с языками низкого уровня.
2. Базовые типы данных, операции над ними, приведение типов.
3. Операторы ветвления: условный оператор, оператор множественного выбора.
4. Операторы цикла: с предусловием, с постусловием, с известным числом повторений.
5. Понятие функции, назначение декомпозиции программы на функции.
6. Способы передачи параметров: по значению, по ссылке, по указателю.
7. Область видимости и время жизни переменных.
8. Понятие рекурсии, сравнение рекурсивного и итеративного подходов.
9. Одномерные и многомерные массивы: объявление, инициализация, обращение к элементам.

10. Понятие указателя, операции над указателями, связь массивов и указателей.
11. Динамическое выделение и освобождение памяти.
12. Типичные ошибки при работе с указателями и динамической памятью.
13. Понятие структуры, объявление структурного типа, обращение к полям.
14. Понятие объединения, отличие от структуры.
15. Организация ввода-вывода данных в текстовые и бинарные файлы.
16. Обработка ошибок при работе с файлами.

7.3.2 Список вопросов к экзамену (6 семестр)

1. Понятие объектно-ориентированного программирования, основные принципы.
2. Понятие класса и объекта, модификаторы доступа к членам класса.
3. Конструкторы и деструкторы, их назначение и особенности вызова.
4. Конструктор копирования, перегрузка конструкторов.
5. Понятие наследования, отношение «является» между классами.
6. Одиночное и множественное наследование.
7. Понятие полиморфизма, статическое и динамическое связывание.
8. Виртуальные функции, механизм их вызова.
9. Абстрактные классы и чисто виртуальные функции.
10. Понятие обобщенного программирования, назначение шаблонов функций и классов.
11. Синтаксис объявления шаблонов, инстанцирование шаблонов компилятором.
12. Обзор стандартной библиотеки шаблонов: контейнеры, итераторы, алгоритмы.
13. Применение стандартных контейнеров для решения типовых задач обработки данных.
14. Понятие исключительной ситуации, механизм обработки исключений.
15. Иерархия исключений, разработка пользовательских классов исключений.
16. Этапы разработки комплексного программного проекта.
17. Особенности разработки программного обеспечения на языке высокого уровня для устройств умных пространств.