МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Национальный исследовательский ядерный университет «МИФИ»
Обнинский институт атомной энергетики –
филиал федерального государственного автономного образовательного учреждения высшего образования
«Национальный исследовательский ядерный университет «МИФИ»
(ИАТЭ НИЯУ МИФИ)

ОТДЕЛЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ КИБЕРНЕТИЧЕСКИХ СИСТЕМ

Одобрено на заседании Ученого
совета ИАТЭ НИЯУ МИФИ,
протокол №25.8 от 25.08.2025

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

название дисциплины

для студентов направления подготовки

38.03.05 Бизнес-информатика

образовательная программа:

Цифровая трансформация компании

Форма обучения: очная

г. Обнинск 2025 г.

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения ООП бакалавриата обучающийся должен овладеть следующими результатами обучения по дисциплине:

Код компетенции	Наименование компетенции	Код и наименование индикатора достижения компетенции
ОПК-3	Способен управлять процессами создания и использования продуктов и услуг в сфере ИКТ, в том числе разрабатывать алгоритмы и программы для их практической реализации	З-ОПК-3 Знать: Методы оценки объемов и сроков выполнения работ Технологии выполнения работ в организации Технологии межличностной и групповой коммуникации в деловом взаимодействии, основы конфликтологии Архитектура, устройство и функционирование вычислительных систем Коммуникационное оборудование Сетевые протоколы Основы современных операционных систем Основы современных систем управления базами данных Устройство и функционирование современных ИС Теория баз данных Системы хранения и анализа баз данных Основы программирования Современные объектно-ориентированные языки программирования Современные структурные языки программирования Языки современных бизнес-приложений Современные методики тестирования разрабатываемых ИС Современные стандарты информационного взаимодействия систем Программные средства и платформы инфраструктуры информационных технологий организаций Современные подходы и стандарты автоматизации организации (например, Web-системы, CRM, MRP, ERP..., ITIL, ITSM) Системы классификации и кодирования информации, в том числе присвоение кодов документам и элементам

Код компетенции	Наименование компетенции	Код и наименование индикатора достижения компетенции
		справочников Отраслевая нормативная техническая документация Источники информации, необходимой для профессиональной деятельности У-ОПК-3 Уметь: Разрабатывать документы Оценивать объемы работ и сроки их выполнения Проводить переговоры В-ОПК-3 Владеть навыками: Подготовка частей коммерческого предложения заказчику касательно объема и сроков выполнения работ по созданию (модификации) и вводу в эксплуатацию типовой ИС Осуществление инженерно-технологической поддержки в ходе согласования коммерческого предложения с заказчиком

2. Место дисциплины в структуре ООП бакалавриата

Дисциплина относится к обязательной части.

Для освоения дисциплины необходимы компетенции, сформированные в рамках изучения следующих дисциплин: «Программирование».

Дисциплина изучается на 2 курсе в 4 семестре.

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

Общая трудоемкость (объем) дисциплины составляет 5 зачетных единиц (з.е.), 180 академических часов.

3.1. Объём дисциплины по видам учебных занятий (в часах)

Вид учебной работы	Всего часов
Контактная работа обучающихся с преподавателем	
Аудиторные занятия *(всего)*	64
В том числе:	
Лекции	16

Практические занятия	16
Лабораторные занятия	32
Промежуточная аттестация	
В том числе:	
экзамен	36
Самостоятельная работа	
Самостоятельная работа обучающихся *(всего)*	80
ОБЩАЯ ТРУДОЕМКОСТЬ	
час	180
зач.ед.	5

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

4.1. Разделы дисциплины и трудоемкость по видам учебных занятий (в академических часах)

Раздел, тема программы учебной дисциплины	Трудоемкость (час)			
	Всего	в том числе по видам учебных занятий		
		лекции	семинары, практические занятия	лабораторные занятия
1	**2**	**3**	**4**	**5**
Раздел 1. Введение в программирование на C++	12	3	0	9
Тема 1. Введение в программирование на C++. Структура программного проекта. Структурирование и оформление кода. Константы, переменные, операции. Классы памяти. Основные управляющие конструкции языка. Массивы, строки и указатели. Структуры и объединения. Функции и передача аргументов. Препроцессор. Макросы. Динамическая память. Ввод/вывод данных. Работа с	12	3	0	9

Раздел, тема программы учебной дисциплины	Трудоемкость (час)			
	Всего	в том числе по видам учебных занятий		
		лекции	семинары, практические занятия	лабораторные занятия
1	**2**	**3**	**4**	**5**
файловой системой. Перегруженные функции. Ссылочные типы.				
Раздел 2. Основы объектно-ориентированного программирования	12	4	0	8
Тема 1. Определение класса. Абстракция. Инкапсуляция. Наследование. Полиморфизм. Объекты классов: статические, автоматические, динамические. Управление доступом к элементам классов. Шаблоны функций. Шаблоны классов. Примеры.	3	1	0	2
Тема 2. Наследование. Множественное наследование. Виртуальные функции. Абстрактные классы. Виртуальная схема наследования. Конструкторы. Деструкторы. Ссылки на объекты классов. Примеры.	3	1	0	2
Тема 3. Полиморфные структуры данных. Списки, деревья, hash-таблицы. Ассоциативные массивы. Контейнеры и коллекции объектов. Примеры.	3	1	0	2
Тема 4. Перегруженные операции (унарные, бинарные). Объекты-исключения. Обработка нештатных ситуаций. Управление	3	1	0	2

Раздел, тема программы учебной дисциплины	Трудоемкость (час)			
	Всего	в том числе по видам учебных занятий		
		лекции	семинары, практические занятия	лабораторные занятия
1	**2**	**3**	**4**	**5**
событиями. Стандартные библиотеки классов MFC, STL. Примеры.				
Раздел 3. Основы объектно-ориентированного проектирования	14	5	10	9
Тема 1. Основы объектно-ориентированного проектирования (UML). Объектный анализ и декомпозиция предметной области. Диаграммы прецедентов, диаграммы классов как способ визуального моделирования ПО. Примеры.	14	5	10	9
Раздел 4. Введение в программирование на C# и Java	12	4	6	8
Тема 1. Введение в программирование на C# и Java. Структура программного проекта. Машинно-независимый код. Интерфейсы и делегаты. Многопоточность. Обертки и рефлексия. Виды приложений: апплеты, сервлеты, портлеты, мидлеты, виджеты. Основные отличия от C++. Примеры.	12	4	6	8
Итого часов	**64**	16	16	32
Аудиторных часов	**64**	Формы рубежного (итогового) контроля знаний – Экзамен		
Внеаудиторная самостоятельная работа	**80**			

Раздел, тема программы учебной дисциплины	Трудоемкость (час)			
	Всего	в том числе по видам учебных занятий		
		лекции	семинары, практические занятия	лабораторные занятия
1	**2**	**3**	**4**	**5**
Количество часов на подготовку к зачету/экзамену	**36**			
Всего часов на освоение учебного материала	**180**			

4.2. Содержание дисциплины, структурированное по разделам (темам)

№ п/п	Наименование раздела дисциплины	Содержание раздела
1.	Введение в программирование на C++	Введение в программирование на C++. Структура программного проекта. Структурирование и оформление кода. Константы, переменные, операции. Классы памяти. Основные управляющие конструкции языка. Массивы, строки и указатели. Структуры и объединения. Функции и передача аргументов. Препроцессор. Макросы. Динамическая память. Ввод/вывод данных. Работа с файловой системой. Перегруженные функции. Ссылочные типы.
2.	Основы объектно-ориентированного программирования	Определение класса. Абстракция. Инкапсуляция. Наследование. Полиморфизм. Объекты классов: статические, автоматические, динамические. Управление доступом к элементам классов. Шаблоны функций. Шаблоны классов. Примеры. Наследование. Множественное наследование. Виртуальные функции. Абстрактные классы. Виртуальная схема наследования. Конструкторы. Деструкторы. Ссылки на объекты классов. Примеры. Полиморфные структуры данных. Списки, деревья, hash-таблицы. Ассоциативные массивы. Контейнеры и коллекции объектов. Примеры. Перегруженные операции (унарные, бинарные). Объекты-исключения. Обработка нештатных ситуаций. Управление событиями. Стандартные библиотеки классов MFC, STL. Примеры.
3.	Основы объектно-ориентированного проектирования	Основы объектно-ориентированного проектирования (UML). Объектный анализ и декомпозиция предметной области. Диаграммы прецедентов, диаграммы классов как способ визуального моделирования ПО. Примеры.
4.	Введение в программирование на C# и Java	Введение в программирование на C# и Java. Структура программного проекта. Машинно-независимый код. Интерфейсы и делегаты. Многопоточность. Обертки и рефлексия. Виды приложений: апплеты, сервлеты, портлеты, мидлеты, виджеты. Основные отличия от C++. Примеры.

Практические занятия (семинары)

№ п/п	№ раздела дисциплины	Тематика практических занятий (семинаров)	Трудо-емкость (час.)
1.	Раздел 3. Основы объектно-ориентированного проектирования	1. Проектирование ПО. Разработка спецификации требований. 2. Проектирование ПО средствами *UML*. Разработка диаграмм прецедентов. 3. Проектирование ПО средствами *UML*. Разработка диаграмм классов. 4. Проектирование ПО средствами *UML*. Разработка диаграмм: состояний, последовательности, кооперации. 5. Проектирование ПО средствами *UML*. Разработка диаграмм: компонентов, развертывания.	10
2.	Раздел 4. Введение в программирование на C# и Java	1. Освоение паттернов проектирования ПО и архитектуры *MVC*. 2. Изучение техники создания компонентных приложений в *.NET Framework* на языке *C#*. 3. Изучение техники создания компонентных приложений в *NetBeans IDE* на языке *Java*.	7

Лабораторные занятия

№ п/п	№ раздела дисциплины	Наименование лабораторных работ	Трудо-емкость (час.)
1.	Раздел 1. Введение в программирование на C++	Лабораторная работа 1. Освоение среды программирования MS Visual Studio. Реализация простейшего индивидуального проекта на языке Си. Варианты заданий: реализовать консольное приложение для заданной динамической структуры данных (вектор, список, стек, дек, очередь, бинарное дерево, hash-таблица, множество, карта, словарь).	9
2.	Раздел 2. Основы объектно-ориентированного программирования	Лабораторная работа 2. Освоение стандартных библиотек классов MFC, Windows Forms, STL и методов управления событиями. Реализация индивидуального объектно-ориентированного проекта на языке C++ в среде MS Visual Studio. Варианты заданий: создать полиморфный контейнер объектов с оконным интерфейсом на базе динамической структуры данных из лабораторной работы 1.	8
3.	Раздел 3. Основы объектно-ориентированного проектирования	Лабораторная работа 3. Освоение базовых средств объектно-ориентированного проектирования. Реализация коллективного (2-4 чел.) проекта на языке C# в среде Visual Paradigm + MS Visual Studio. Варианты заданий: создать действующую программную модель простой системы или процесса реального мира. Список предметных областей для моделирования прилагается.	9

№ п/п	№ раздела дисциплины	Наименование лабораторных работ	Трудо-емкость (час.)
4.	Раздел 4. Введение в программирование на C# и Java	Лабораторная работа 4. Освоение среды программирования NetBeans IDE и объектно-ориентированной библиотеки Swing. Индивидуальная разработка простейшего апплета на языке Java. Варианты заданий: реализовать в окне браузера программное решение, ранее разработанное в лабораторной работе 2 или 3.	8

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

5.1. Программное обеспечение

Microsoft Visual Studio.

5.2. Электронные ресурсы:

1. http://vt.obninsk.ru
2. http://msdn.microsoft.com/ru-ru/library/default.aspx
3. http://www.visual-paradigm.com

Материалы учебно-методического комплекса дисциплины в полном объеме доступны на образовательном портале «Кафедра онлайн», который размещается по сетевому адресу http://vt.obninsk.ru, см. компонент «Облачный кабинет».

5.3. Домашние тренинговые задания

Домашние тренинговые задания №1

1. Опишите следующие объекты: указатель на символ; массив из 10 целых; ссылка на массив из 10 целых; указатель на массив символьных строк; указатель на указатель на символ; целое-константа; указатель на целое-константу; константный указатель на целое. Описания снабдите инициализацией.
2. Разработайте функцию, которая печатает размеры основных типов и типа указателя. Используйте операцию sizeof.
3. Разработайте функцию, которая выводит буквы от 'a' до 'z' и цифры от '0' до '9' и значения их кодов. Проделайте то же самое для других видимых символов. Проделайте это, используя шестнадцатеричную запись.

4. Разработайте функцию, которая выводит порядок и мантиссу своего аргумента типа double.
5. Составьте таблицу из названий месяцев года и числа дней в каждом из них. Разработайте функцию, которая выводит эту таблицу. Проделайте это дважды: один раз - используя массивы для названий месяцев и количества дней, а другой раз - используя массив структур, каждая из которых содержит название месяца и количество дней в нем.

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

6.1. Паспорт фонда оценочных средств по дисциплине

№ п/п	Контролируемые модули, разделы дисциплины	Индикатор достижения компетенции	Наименование оценочного средства
1	Раздел 1. Введение в программирование на C++	З-ОПК-3; У-ОПК-3; В-ОПК-3	Письменно. Лабораторная работа №1. Письменно.
2	Раздел 2. Основы объектно-ориентированного программирования	З-ОПК-3; У-ОПК-3; В-ОПК-3	Контрольная работа №1. Лабораторная работа №2. Письменно.
3	Раздел 3. Основы объектно-ориентированного проектирования	З-ОПК-3; У-ОПК-3; В-ОПК-3	Контрольная работа №2. Лабораторная работа №3. Письменно.
4	Раздел 4. Введение в программирование на C# и Java	З-ОПК-3; У-ОПК-3; В-ОПК-3	Лабораторная работа №4. Письменно.

6.2. Типовые контрольные задания или иные материалы

6.2.1. Экзамен

Федеральное государственное автономное образовательное учреждение высшего профессионального образования

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ УНИВЕРСИТЕТ «МИФИ»

ОБНИНСКИЙ ИНСТИТУТ АТОМНОЙ ЭНЕРГЕТИКИ

Дисциплина «Объектно-ориентированное программирование»

Билет № 1

Вопрос №1

Основные операции языка Си.

Вопрос №2

Язык Си: низкоуровневый ввод/вывод. Пример.

Вопрос № 3

Модели жизненного цикла программного обеспечения.

Начальник ОИКС _______ Старков С.О.

Критерии оценки:

Оценка,%	**Характеристики ответа студента**
76 – 100%	– знания отличаются глубиной и содержательностью, даны логично построенные, полные, исчерпывающие ответы на вопросы; – обучающийся демонстрирует способность к анализу практических ситуаций, оперирует профессиональными терминами; – обучающийся умеет высказывать и обосновать свои суждения;
51 – 75%	– знания имеют достаточный содержательный уровень, однако имеются определенные затруднения в ответе на уточняющие вопросы; – несущественные ошибки в определении понятий и категорий и т. п., кардинально не меняющие суть изложения;
26 – 50%	– знания имеют фрагментарный характер, имеются определенные неточности и погрешности в формулировках, возникают затруднения при ответе на уточняющие вопросы; – при ответе на вопросы обучающийся не может обосновать закономерности, принципы, объяснить суть явления; допущены фактические ошибки; – обучающийся продемонстрировал слабое умение формулировать выводы и обобщения,
0 – 25%	– обучающийся слабо ориентируется в теме задания, затрудняется дать ответ или дает неверные ответы на поставленные вопросы; – допущены существенные фактические ошибки при ответах на вопросы.

6.2.2. Контрольная работа

Комплект заданий для контрольной работы

Вариант 1

1. Разработайте класс объектов типа **«текстовые файлы на жестком диске, которые способны отображать своё содержимое на экране».** При необходимости приведите схему наследования. Снабдите Ваше решение комментариями.
2. Опишите абстрактный класс объектов, для которых определены операции **==, =, !=.** Операцию == определите полностью для случая, когда объекты являются строками символов.

Вариант 2

1. Разработайте класс объектов типа «текстовые файлы на жестком диске, которые способны распечатывать своё содержимое на принтере». При необходимости приведите схему наследования. Снабдите Ваше решение комментариями.
2. Опишите абстрактный класс объектов, для которых определены операции ==, !=, ~. Операцию ~ определите полностью для случая, когда объекты имеют тип unsigned int.

Вариант 3

1. Разработайте класс объектов типа «строки, которые способны отображать себя в заданном месте экрана заданным шрифтом и цветом». При необходимости приведите схему наследования. Снабдите Ваше решение комментариями.
2. Опишите абстрактный класс объектов, для которых определены операции &&, ||, ~. Операцию ~ определите полностью для случая, когда объекты имеют тип signed char.

Вариант 4

1. Разработайте класс объектов типа «папки (директории) файловой системы Windows, которые способны отображать своё содержимое на экране». При необходимости приведите схему наследования. Снабдите Ваше решение комментариями.
2. Опишите абстрактный класс объектов, для которых определены операции +, -, !=. Операцию != определите полностью для случая, когда объекты являются строками символов.

Вариант 5

1. Разработайте класс объектов типа «прямоугольные графические образы (фотографии), которые способны отображать себя на экране и сохранять на жестком диске». При необходимости приведите схему наследования. Снабдите Ваше решение комментариями.

2. Опишите абстрактный класс объектов, для которых определены операции +=, -=, !. Операцию ! определите полностью для случая, когда объекты имеют тип double.

Критерии оценки:

Оценка,%	**Характеристики ответа студента**
76 – 100%	– знания отличаются глубиной и содержательностью, даны логично построенные, полные, исчерпывающие ответы на вопросы; – обучающийся демонстрирует способность к анализу практических ситуаций, оперирует профессиональными терминами; – обучающийся умеет высказывать и обосновать свои суждения;
51 – 75%	– знания имеют достаточный содержательный уровень, однако имеются определенные затруднения в ответе на уточняющие вопросы; – несущественные ошибки в определении понятий и категорий и т. п., кардинально не меняющие суть изложения;
26 – 50%	– знания имеют фрагментарный характер, имеются определенные неточности и погрешности в формулировках, возникают затруднения при ответе на уточняющие вопросы; – при ответе на вопросы обучающийся не может обосновать закономерности, принципы, объяснить суть явления; допущены фактические ошибки; – обучающийся продемонстрировал слабое умение формулировать выводы и обобщения,
0 – 25%	– обучающийся слабо ориентируется в теме задания, затрудняется дать ответ или дает неверные ответы на поставленные вопросы; – допущены существенные фактические ошибки при ответах на вопросы.

6.2.3. Лабораторная работа

Лабораторная работа 1.

Освоение среды программирования MS Visual Studio. Реализация простейшего индивидуального проекта на языке Си. Варианты заданий: реализовать консольное приложение для заданной динамической структуры данных (вектор, список, стек, дек, очередь, бинарное дерево, hash-таблица, множество, карта, словарь)

Лабораторная работа 2.

Освоение стандартных библиотек классов MFC, Windows Forms, STL и методов управления событиями. Реализация индивидуального объектно-ориентированного проекта на языке C++ в среде MS Visual Studio. Варианты заданий: создать полиморфный контейнер объектов с оконным интерфейсом на базе динамической структуры данных из лабораторной работы 1.

Лабораторная работа 3.

Освоение базовых средств объектно-ориентированного проектирования. Реализация коллективного (2-4 чел.) проекта на языке C# в среде Visual Paradigm + MS Visual Studio. Варианты заданий: создать действующую программную модель простой системы или процесса реального мира. Список предметных областей для моделирования прилагается.

Лабораторная работа 4.
Освоение среды программирования NetBeans IDE и объектно-ориентированной библиотеки Swing. Индивидуальная разработка простейшего апплета на языке Java. Варианты заданий: реализовать в окне браузера программное решение, ранее разработанное в лабораторной работе 2 или 3.

7. Перечень учебной литературы, необходимой для освоения дисциплины

1. Объектно-ориентированное программирование : учебное пособие / В. В. Лозовский, Е. Н. Штрекер, Е. С. Данилович [и др.]. — Москва : РТУ МИРЭА, 2025. — 484 с. — ISBN 978-5-7339-2498-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/493547

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее - сеть «Интернет»), необходимых для освоения дисциплины

1. http://vt.obninsk.ru
2. http://msdn.microsoft.com/ru-ru/library/default.aspx
3. http://www.visual-paradigm.com

Материалы учебно-методического комплекса дисциплины в полном объеме доступны на образовательном портале «Кафедра онлайн», который размещается по сетевому адресу http://vt.obninsk.ru, см. компонент «Облачный кабинет».

9. Методические указания для обучающихся по освоению дисциплины

В ходе лабораторного практикума студенты в составе малых проектных групп (3 – 4 чел.) выполняют проектные задания (лабораторные работы). В течение семестра последовательно создаются прототипы программного обеспечения (которым соответствуют контрольные точки проекта). Защита лабораторных работ (проектов) происходит в форме деловой игры. Оппонентами проекта выступают участники иных проектных групп, преподаватели и, в отдельных случаях, приглашенные лица.

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

10.1. Перечень информационных технологий

Использование видео- и аудиоматериалов, дистанционного консультирования и вебинаров с применением интернет-портала «Кафедра онлайн», расположенного по сетевому адресу http://vt.obninsk.ru, см. компонент «Учебный форум студентов».

Предоставление учебных web-хостингов, который расположены по сетевому адресу http://vt.obninsk.ru и на облачных сервисах, для размещения, отладки и тестирования студенческих работ по изучаемым дисциплинам.

10.2. Перечень информационных справочных систем

Доступ к электронным библиотечным ресурсам и электронной библиотечной системе (ЭБС) осуществляется посредством специальных разделов на официальном сайте ИАТЭ НИЯУ МИФИ. Обеспечен доступ к электронным каталогам библиотеки ИАТЭ НИЯУ МИФИ, а также электронным образовательным ресурсам (ЭИОС), сформированным на основании прямых договоров с правообладателями учебной и учебно-методической литературы, методических пособий:

1. ЭБС «Айбукс», Договор № 05-25-910 от 2.06.2025 на оказание услуг по предоставлению доступа к электронно-библиотечной системе «Айбукс.ру/ibooks.ru». Срок действия: 1.09.2025г. по 31.08.2026г.
2. ЭБС «ЛАНЬ», Договор № 08-25-910 от 30.05.2025 на оказание услуг по предоставлению доступа к базе данных ЭБС «Лань». Срок действия: с 1.09.2025г. по 31.08.2026г.
3. ЭБС «ЮРАЙТ», Договор № 09-25-910 от 3.06.2025 на оказание услуг по предоставлению доступа к «Образовательной платформе ЮРАЙТ». Срок действия: с 1.09.2025г. по 31.08.2026г.
4. ЭБС «Консультант студента», Договор № 06-25-910 от 3.06.2025 на оказание услуг по предоставлению доступа к комплектам «Медицина, Здравоохранение», «Книги издательства «Феникс», «Издательский дом МЭИ», «Книги издательства «Проспект»: «Иностранные языки», «Естественные науки», «Экономика и управление», «Гуманитарные науки», «Юридические науки», входящим в базу данных «Электронная библиотека технического вуза». (ЭБС «Консультант студента»). Срок действия: 1.09.2025г. по 31.08.2026г.
5. Научная электронная библиотека E-library.ru, Договор № SU-353/2025 от 2.11.2024 на доступ к электронным изданиям в составе базы данных «НАУЧНАЯ ЭЛЕКТРОННАЯ БИБЛИОТЕКА eLIBRARY.RU». Срок действия: с 1.01.2025г. по 31.12.2025г.

11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Специализированные компьютерные классы. 12 компьютеризованных рабочих мест студента.

12. Иные сведения и (или) материалы

12.1. Перечень образовательных технологий, используемых при осуществлении образовательного процесса по дисциплине

В ходе преподавания дисциплины применяются следующие методы интерактивного обучения:

- Метод проектных технологий и малых проектных групп как форма выполнения студентами лабораторных работ (проектных заданий).
- Круглый стол, дискуссия, дебаты как форма консультирования студентов.
- Деловые и ролевые игры как форма организации работы в проектных группах и форма защиты выполненных проектных заданий.
- Мозговой штурм, case-study (коллективный анализ конкретных ситуаций, ситуационный анализ) при поиске вариантов решения задач, сформулированных в проектных заданиях.
- Мастер классы, тренинги и симуляции, которые организуют студенты-магистранты.
- Метод обучения в парах (в частности, парное программирование) в ходе выполнения проектных заданий.

12.2. Формы организации самостоятельной работы обучающихся (темы, выносимые для самостоятельного изучения; вопросы для самоконтроля; типовые задания для самопроверки

1. Основные операции языка Си.
2. Язык Си: переключатели, перечислимые типы. Примеры.
3. Язык Си: указатели и массивы. Определения, примеры.
4. Язык Си: объявления функций, передача аргументов. Примеры.
5. Язык Си: строки и указатели. Определения, примеры.
6. Язык Си: аргументы командной строки. Пример.
7. Язык Си: указатели на функции. Пример.
8. Язык Си: структуры. Пример.
9. Язык Си: битовые поля и объединения. Примеры.
10. Язык Си: оператор определения типа. Примеры.
11. Язык Си: препроцессор. Директивы препроцессора, примеры.
12. Язык Си: программный стек. Пример работы стека.
13. Линейные списки. Операции с линейными списками.
14. Hash-таблицы (таблицы с перемешиванием).
15. Двоичные деревья. Алгоритмы обхода, поиска, вставки, сортировки.
16. Язык Си: перегруженные функции. Пример.

17. Язык Си: ссылочные типы. Пример.
18. Язык Си: защита указателей и объектов, неявное изменение объектов.
19. Язык Си: организация ввода/вывода. Пример: слияние файлов.
20. Язык Си: произвольный доступ к файлам. Пример.
21. Язык Си: низкоуровневый ввод/вывод. Пример.
22. Язык Си: примеры реализаций функций ввода/вывода (getc, putc).
23. Язык Си: примеры реализаций функций ввода/вывода (fgets, fputs).
24. Язык Си: работа с файловой системой. Пример.
25. Основные понятия ООП: абстракция, инкапсуляция, наследование, полиморфизм.
26. Альтернативное определение ООП.
27. Объекты классов: статические, автоматические, динамические. Примеры.
28. Управление доступом к элементам классов. Пример.
29. Шаблоны функций и шаблоны классов. Примеры.
30. Наследование. Пример.
31. Множественное наследование. Пример.
32. Виртуальные функции. Раннее и позднее связывание.
33. Абстрактные классы. Их назначение, пример.
34. Полиморфный контейнер (пример).
35. Виртуальная схема наследования.
36. Конструкторы, их виды, примеры. Вызов конструкторов при наследовании.
37. Деструкторы. Их назначение, примеры.
38. Перегруженные операции. Формализм на языке C++.
39. Перегруженные операции. Примеры.
40. Обработка нештатных ситуаций. Объекты-исключения. Примеры.
41. Модели жизненного цикла программного обеспечения.
42. Проектирование программного обеспечения и UML.
43. Диаграммы прецедентов. Нотация, семантика, примеры.
44. Сценарии выполнения прецедентов (пример).
45. Диаграммы классов. Нотация, семантика, отношения.
46. Атрибуты на диаграммах классов. Нотация и семантика. Примеры.
47. Операции на диаграммах классов. Нотация и семантика. Примеры.
48. Отношения ассоциации на диаграммах классов.
49. Отношения обобщения на диаграммах классов.
50. Отношения агрегации на диаграммах классов.
51. Отношения композиции на диаграммах классов.
52. Обзор платформы .NET Framework. C# как язык ООП для .NET Framework.
53. Язык C#: сборки, манифесты, домены, компоненты.
54. Обзор различий между языками C# и C++.
55. Java как язык ООП для машинно-назависимых приложений.
56. Обзор Java-технологий NetBeans IDE.
57. Компонентные технологии разработки программного обеспечения.
58. Обзор различий между языками Java и C++.

59. Обёртки в языках C# и Java.
60. Архитектурный паттерн MVC. Область применения, схема взаимодействия.

12.3. Краткий терминологический словарь

Класс является расширением понятия типа и позволяет объединить данные и код для их обработки. Он подходит для описания реальных объектов и их абстракций, где элементы данных являются свойствами объекта, а функции или методы описывают поведение объекта.

Классы содержат особые методы называемые конструкторами и деструкторами. Первые предназначены для инициализации создаваемых объектов, вторые вызываются при уничтожении объекта.

Инкапсуляция - сокрытие реализации класса от его пользователя. Другими словами, пользователь имеет доступ только к открытым членам класса. К закрытым членам имеет доступ только сам объект. Это позволяет избежать ошибки, связанные с нечаянным изменением членов данных класса.

Наследование - механизм позволяющий описать новый класс на основе уже существующего класса. Новый класс называют дочерним классом, потомком, подклассом или производным классом по отношению к родительскому или базовому классу.

Полиморфизм - механизм позволяющий изменить реализацию методов, т.е. поведение объекта в дочерних классах. Изменяемые методы называются виртуальными. В частности, если класс содержит объявление метода, реализация которого предполагается только в дочерних классах, то он называется абстрактным классом.