

**Автономная некоммерческая организация высшего образования
«МОСКОВСКИЙ МЕЖДУНАРОДНЫЙ УНИВЕРСИТЕТ»**

Рабочая программа дисциплины

Алгоритмизация и программирование

<i>Направление подготовки</i>	Прикладная информатика
<i>Код</i>	09.03.03
<i>Направленность (профиль)</i>	Прикладная информатика в бизнесе
<i>Квалификация выпускника</i>	бакалавр

Москва
2026

1. Перечень кодов компетенций, формируемых дисциплиной в процессе освоения образовательной программы

Группа компетенций	Категория компетенций	Код
Общепрофессиональные	-	ОПК-2
Общепрофессиональные	-	ОПК-3
Общепрофессиональные	-	ОПК-4
Общепрофессиональные	-	ОПК-7

2. Компетенции и индикаторы их достижения

Код компетенции	Формулировка компетенции	Индикаторы достижения компетенции
ОПК-2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1 Знает принципы работы современных информационных технологий и программных средств, в том числе отечественного производства при решении задач профессиональной деятельности ОПК-2.2 Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности ОПК-2.3 Владеет навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности
ОПК-3	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	ОПК-3.1 Знает принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности ОПК-3.2 Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности ОПК-3.3 Владеет навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций, и

		библиографии по научно-исследовательской работе с учетом требований информационной безопасности
ОПК-4	Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	ОПК-4.1 Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.2 Умеет применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.3 Владеет навыками составления технической документации на различных этапах жизненного цикла информационной системы
ОПК-7	Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1 Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий ОПК-7.2 Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ ОПК-7.3 Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач

3. Описание планируемых результатов обучения по дисциплине и критериев оценки результатов обучения по дисциплине

3.1. Описание планируемых результатов обучения по дисциплине

Планируемые результаты обучения по дисциплине представлены дескрипторами (знания, умения, навыки)

Дескрипторы по дисциплине	Знать	Уметь	Владеть
Код компетенции	ОПК-2		
	- особенности и ограничения применимости инструментов для известного	- обоснованно выбрать необходимые технологии для решения	- организации межпрограммного взаимодействия для решения прикладных задач конечного

	контекста	поставленной задачи	пользователя; систематизации программного обеспечения
Код компетенции	ОПК-3		
	- классификацию и критерии классификации информационных технологий; - характеристики базовых информационных процессов сбора, передачи, обработки, хранения и представления информации, а также средства реализации базовых информационных процессов	- работать с информацией в глобальных компьютерных сетях и корпоративных информационных системах; - использовать системы поиска профессиональной информации в глобальных сетях	- работы в локальных и глобальных сетях при решении научных и исследовательских задач; -использования программных средств обеспечения безопасности данных на автономном ПК и в интерактивной среде
Код компетенции	ОПК-4		
	- основные принципы процесса разработки программного обеспечения; - методы и средства разработки программной документации	- формализовать прикладную задачу, выбирать для неё подходящие структуры данных и алгоритмы обработки; - пользоваться основными методологиями процессов разработки программного обеспечения и документации	- навыками применения методов и средств разработки программной документации
Код компетенции	ОПК-7		
	- принципы объектно-ориентированного программирования; - языки и среды программирования; - основные этапы разработки программного обеспечения	- программировать алгоритмы, используя средства языков программирования; - создавать программу по разработанному алгоритму как отдельный модуль	- проведения тестирования программного модуля по определенному сценарию

4. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина относится к обязательной части учебного плана образовательной программы. Данная дисциплина взаимосвязана с другими дисциплинами, такими как «Интернет-программирование», «Программная инженерия», «Проектирование информационных систем».

В рамках освоения программы бакалавриата выпускники готовятся к решению задач профессиональной деятельности следующих типов: производственно-технологический, научно-исследовательский и проектный.

Профиль (направленность) программы установлена путем её ориентации на сферу профессиональной деятельности выпускников.

5. Объем дисциплины

<i>Виды учебной работы</i>	<i>Форма обучения</i>
	<i>Очная</i>
Общая трудоемкость: зачетные единицы/часы	9/324
Контактная работа:	
Занятия лекционного типа	72
Занятия семинарского типа	108
Промежуточная аттестация: зачет, зачет с оценкой, экзамен	9,35
Самостоятельная работа (СРС)	134,65

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

6.1. Распределение часов по разделам/темам и видам работы

6.1.1. Очная форма обучения

№ п/п	Раздел/тема	Виды учебной работы (в часах)						
		Контактная работа						Самост оательн ая работа
		Занятия лекционного типа		Занятия семинарского типа				
		Лекции	Иные учебны е заняти я	Практ ически е заняти я	Семин ары	Лабор аторн ые работ ы	Иные	
1.	Алгоритмы. Способы представления алгоритмов	2		3				4
2.	Базовые элементы языка программирования	2		3				4
3.	Средства разработки программ	2		3				4
4.	Базовые конструкции структурного	2		3				4

	программирования							
5.	Функции	2		3				4
6.	Массивы данных. Алгоритмы сортировки данных	2		3				4
7.	Указатели и ссылки. Динамическое распределение памяти	2		3				4
8.	Типы данных, определяемые пользователем	2		3				4
9.	ООП. Классы и объекты.	2		3				4
10.	Отношение наследования	2		3				4
11.	Механизмы полиморфизма	2		3				4
12.	Шаблоны функций	2		3				4
13.	Библиотека шаблонов	2		3				4
14.	Потоковые классы. Файлы	2		3				4
15.	Строки.	2		3				4
16.	Последовательны е и ассоциативные контейнеры	2		3				4
17.	Коллекции	2		3				4
18.	Обработка исключительных ситуаций	2		3				4
19.	Работа с файловой системой	2		3				4
20.	Классы. Отношения вложенности	2		3				4
21.	Методы класса с особым синтаксисом	2		4				4
22.	Интерфейсы	2		4				4
23.	Делегаты	2		4				4
24.	События	2		4				4
25.	Построение приложений с графическим интерфейсом	2		4				4
26.	Диалоговые окна	2		4				4,65
27.	Источники данных. Привязка к источнику	2		4				5

	данных							
28.	Взаимодействие форм приложения	2		4				5
29.	Способы табличного отображения данных	4		4				5
30.	Способы подключения к базам данных	4		4				5
31.	Язык интегрированных запросов LINQ и элементы функционального программирования	4		4				5
32.	Многопоточность. Параллельное программирование	4		4				5
	Промежуточная аттестация	9,35						
	Итого	72		108				134,65

6.2.Программа дисциплины, структурированная по темам / разделам

6.2.1. Содержание лекционного курса

№ п/п	Наименование темы (раздела) дисциплины	Содержание лекционного занятия
1.	Алгоритмы. Способы представления алгоритмов	Алгоритм. Свойства алгоритмов. Основные этапы разработки алгоритмов: постановка задачи, построение математической модели, разработка алгоритма решения задачи, проверка правильности и оценка сложности алгоритма. Формы и способы представления алгоритмов. Базовые алгоритмические конструкции. Вспомогательные алгоритмы. Алгоритмически неразрешимые проблемы.
2.	Базовые элементы языка программирования	Классификация языков программирования. Синтаксис и семантика языка программирования. Формальные методы описания синтаксиса. Алфавит. Идентификаторы. Ключевые слова. Типы данных. Классификация типов. Конверсия типов. Переменные. Константы. Операции. Выражения. Классификация и приоритет операций. Операторы. Комментарии.
3.	Средства разработки программ	Интегрированная среда разработки. Архитектурные типы. Основные компоненты: поиск и навигация, редактирование и рефакторинг, отладка и тестирование. Система управления версиями.
4.	Базовые конструкции структурного программирования	Теорема Бёма — Якопини. Структура программы. Комментарии. Стандартный ввод/вывод. Конструкция следования. Стандартные функции языка. Организация ветвлений: бинарное и множественное ветвление. Тернарная операция. Множественный выбор. Организация циклов. Вложенные циклы. Принудительное завершение итерации и цикла.

5.	Функции	Описание и определение функций. Вызов функции. Возвращаемое значение функции. Параметры функции: позиционные, по умолчанию, именованные. Функция с переменным числом параметров. Передача параметров по значению и по адресу. -функции. Использование глобальных переменных. Рекурсивная функция. Перегрузка функций. Правила описания перегруженных функций. Схема алгоритма поиска функции.
6.	Массивы данных. Алгоритмы сортировки данных	Одномерные и многомерные массивы. Определение и инициализация. Доступ к элементу массива. Пакетная обработка массива. Линейный и бинарный поиск в массиве. Модификация массивов. Передача массивов в функцию. Массивы символов. Сортировка. Классификация сортировок. Основные виды сортировок: пузырьковая, выбором, шейкерная, расческой. Эффективные сортировки: быстрая, слиянием.
7.	Указатели и ссылки. Динамическое распределение памяти	Указатели. Виды указателей. Способы инициализации указателей. Операции с указателями. Арифметические операции с указателями. Ссылки. Правила инициализации ссылок. Виды ссылок. Ссылки на массив и на функцию. Динамически распределяемая память. Выделение динамической памяти: оператор new. Освобождение динамической памяти: оператор delete. Указатели и динамическая память. Массивы в динамической памяти. Передача динамических массивов в функции. Отслеживание утечек памяти.
8.	Типы данных, определяемые пользователем	Перечисляемый тип. Структура. Инициализация структуры. Элементы структуры. Доступ к полям структуры. Битовые поля. Объединения. Ограничения накладываемые на объединения.
9.	ООП. Классы и объекты.	ООП. Классы. Описание класса. Члены класса. Управление доступом к членам класса. Экземпляры класса. Доступ к элементам экземпляра. Конструкторы. Указатель this. Статические элементы класса. Расширение интерфейса класса. Деструкторы
10.	Отношение наследования	Наследование. Базовые и производные классы. Ключи доступа. Одиночное наследование. Прямое и косвенное наследование. Правила наследования членов класса. Замещение метода базового класса. Иерархия классов.
11.	Механизмы полиморфизма	Полиморфное поведение. Раннее и позднее связывание. Виртуальные методы. Правила использования виртуальных методов. Абстрактные классы. Перегрузка операций.
12.	Шаблоны функций	Шаблоны функций. Шаблонная функция. Шаблонные параметры. Инстанцирование шаблона. Специализация шаблона. Перегрузка шаблонов функции. Шаблоны класса. Правила описания шаблонов класса. Шаблоны как параметры шаблона класса
13.	Библиотека шаблонов	Стандартная библиотека шаблонов. Итераторы. Категории итераторов. Алгоритмы. Категории

		алгоритмов. Функции-предикаты: унарные и бинарные предикаты. Стандартные функциональные объекты. Свойства функциональных объектов. Собственные функциональные объекты.
14.	Потоковые классы. Файлы	Поток. Стандартные, файловые и строковые потоки. Использование файловых потоков для работы с файлами. Текстовые и бинарные файлы. Режимы открытия файла. Операции с файлом: создание, открытие, закрытие файла, чтение или запись. Функции для работы с файлами.
15.	Строки.	Конструкторы. Операции над строками. Доступ к элементу строки. Функции для работы со строками: присваивание и добавление частей строк, получение характеристик строк, преобразование строк, поиск подстрок, сравнение частей строк.
16.	Последовательные и ассоциативные контейнеры	Последовательные контейнеры. Адаптеры последовательных контейнеров. Базовые методы и характерные операции контейнеров. Ассоциативные контейнеры. Общие методы контейнеров.
17.	Коллекции	Индексируемая коллекция. Класс Array. Одномерные и многомерные массивы: объявление и инициализация. Класс List. Создание списка. Добавление и удаление элементов. Статические методы поиска и сортировки в индексируемых коллекциях. Обход коллекции с помощью оператора foreach. Коллекции пар "ключ-значение". Класс Dictionary. Создание словаря. Добавление пары ключ-значение. Обход элементов словаря. Поиск элементов по ключу. Строки как коллекции. Класс String. Неизменяемость строки. Основные статические и динамические методы класса String. Операции над строками.
18.	Обработка исключительных ситуаций	Исключение. Подсистема обработки исключений. Контролируемый блок кода. Обработчики исключений. Блок завершения. Генерация исключения. Перехват исключений. Класс Exception. Определение собственных классов исключений. Иерархия исключений.
19.	Работа с файловой системой	Класс Stream. Работа с текстовыми и бинарными файлами с помощью потоков: классы StreamReader и StreamWriter, BinaryReader и BinaryWriter. Сериализация. Кодировки текстовых файлов. Проверка наличия каталога и файла.
20.	Классы. Отношения вложенности	Клиенты и поставщики. Варианты реализации отношения "клиент-поставщик": создание объекта класса поставщика, вызов статических полей или методов класса поставщика. Отношения между клиентами и поставщиками.
21.	Методы класса с особым синтаксисом	Свойства. Определение свойства. Синтаксис автосвойств. Управление доступом для методов set и get. Вычисляемые свойства. Индексаторы. Объявления индексаторов. Сценарии использования индексаторов.

22.	Интерфейсы	Множественное наследование. Интерфейсы. Наследование интерфейсов. Отличие абстрактных классов и интерфейсов. Неявная реализация интерфейсов. Явная реализация: обертывание и кастинг. Коллизия имен интерфейсов. Склеивание и переименование методов. Встроенные интерфейсы.
23.	Делегаты	Делегаты. Свойства делегата. Объявление делегата. Экземпляр делегата. Одноадресный и многоадресный делегат. Механизм работы с делегатом. Использование лямбда-выражений.
24.	События	События. Свойства событий. Издатели и подписчики. Определение и вызов событий. Обработчики события. Подписка на события. Подписка на события с помощью анонимной функции. Отмена подписки на события.
25.	Построение приложений с графическим интерфейсом	Формы. Элементы управления. Группы элементов управления. Класс Control: свойства и методы. События класса Control. Параметры обработчика события. Программное управление интерфейсом. Группирование элементов. Главное меню. Контекстное меню. Панель инструментов. Строка состояния
26.	Диалоговые окна	Диалоговое окно MessageBox. Перечисления MessageBoxButtons и MessageBoxIcon. Идентификация нажатой кнопки: перечисление DialogResult. Стандартные диалоговые окна: OpenFileDialog, SaveFileDialog, FolderBrowserDialog, ColorDialog.
27.	Источники данных. Привязка к источнику данных	Источники данных. Простая привязка. Сложная привязка. Свойство DataSource. Компонент BindingSource. Режим визуальной привязки. Обновление привязки и перерисовка элемента управления. BindingNavigator.
28.	Взаимодействие форм приложения	Родительские и дочерние формы. Обмен данными между формами приложения. Стандартные диалоговые окна. Всплывающее окно. Управление формами в Main. Немодальные окна. Приложения с интерфейсом MDI.
29.	Способы табличного отображения данных	Элемент управления DataGridView. Коллекции объектов таблицы: столбцы, строки, ячейки. Способы загрузки данных в DataGridView: привязка к источнику данных, программным способом.
30.	Способы подключения к базам данных	Библиотеки для работы с базами данных. Поставщики данных. Строка подключения. Последовательность операций для доступа к данным. Представление данных с помощью DataGridView. Операции CRUD. Создание моделей.
31.	Язык интегрированных запросов LINQ и элементы функционального программирования	Механизм доступа к данным LINQ. LINQ при работе с массивами. SQL-стиль использования LINQ. Использование LINQ через методы. Использование LINQ для доступа к XML. Использование LINQ для работы с базами данных
32.	Многопоточность. Параллельное программирование	Параллельная программа. Класс Thread. Передача параметров в поток. Конкурентный доступ. Пул потоков. Ключевые слова async/await. Асинхронное

		программирование. Асинхронные методы. Типы возвращаемых значений
--	--	--

6.2.2. Содержание практических занятий

№ п/п	Наименование темы (раздела) дисциплины	Содержание практического занятия
1.	Алгоритмы. Способы представления алгоритмов	Алгоритм. Свойства алгоритмов. Основные этапы разработки алгоритмов: постановка задачи, построение математической модели, разработка алгоритма решения задачи, проверка правильности и оценка сложности алгоритма. Формы и способы представления алгоритмов. Базовые алгоритмические конструкции. Вспомогательные алгоритмы. Алгоритмически неразрешимые проблемы.
2.	Базовые элементы языка программирования	Классификация языков программирования. Синтаксис и семантика языка программирования. Формальные методы описания синтаксиса. Алфавит. Идентификаторы. Ключевые слова. Типы данных. Классификация типов. Конверсия типов. Переменные. Константы. Операции. Выражения. Классификация и приоритет операций. Операторы. Комментарии.
3.	Средства разработки программ	Интегрированная среда разработки. Архитектурные типы. Основные компоненты: поиск и навигация, редактирование и рефакторинг, отладка и тестирование. Система управления версиями.
4.	Базовые конструкции структурного программирования	Теорема Бёма — Якопини. Структура программы. Комментарии. Стандартный ввод/вывод. Конструкция следования. Стандартные функции языка. Организация ветвлений: бинарное и множественное ветвление. Тернарная операция. Множественный выбор. Организация циклов. Вложенные циклы. Принудительное завершение итерации и цикла.
5.	Функции	Описание и определение функций. Вызов функции. Возвращаемое значение функции. Параметры функции: позиционные, по умолчанию, именованные. Функция с переменным числом параметров. Передача параметров по значению и по адресу. -функции. Использование глобальных переменных. Рекурсивная функция. Перегрузка функций. Правила описания перегруженных функций. Схема алгоритма поиска функции.
6.	Массивы данных. Алгоритмы сортировки данных	Одномерные и многомерные массивы. Определение и инициализация. Доступ к элементу массива. Пакетная обработка массива. Линейный и бинарный поиск в массиве. Модификация массивов. Передача массивов в функцию. Массивы символов. Сортировка. Классификация сортировок. Основные виды сортировок: пузырьковая, выбором, шейкерная, расческой. Эффективные сортировки: быстрая, слиянием.
7.	Указатели и ссылки.	Указатели. Виды указателей. Способы инициализации

	Динамическое распределение памяти	указателей. Операции с указателями. Арифметические операции с указателями. Ссылки. Правила инициализации ссылок. Виды ссылок. Ссылки на массив и на функцию. Динамически распределяемая память. Выделение динамической памяти: оператор new. Освобождение динамической памяти: оператор delete. Указатели и динамическая память. Массивы в динамической памяти. Передача динамических массивов в функции. Отслеживание утечек памяти.
8.	Типы данных, определяемые пользователем	Перечисляемый тип. Структура. Инициализация структуры. Элементы структуры. Доступ к полям структуры. Битовые поля. Объединения. Ограничения накладываемые на объединения.
9.	ООП. Классы и объекты.	ООП. Классы. Описание класса. Члены класса. Управление доступом к членам класса. Экземпляры класса. Доступ к элементам экземпляра. Конструкторы. Указатель this. Статические элементы класса. Расширение интерфейса класса. Деструкторы
10.	Отношение наследования	Наследование. Базовые и производные классы. Ключи доступа. Одиночное наследование. Прямое и косвенное наследование. Правила наследования членов класса. Замещение метода базового класса. Иерархия классов.
11.	Механизмы полиморфизма	Полиморфное поведение. Раннее и позднее связывание. Виртуальные методы. Правила использования виртуальных методов. Абстрактные классы. Перегрузка операций.
12.	Шаблоны функций	Шаблоны функций. Шаблонная функция. Шаблонные параметры. Инстанцирование шаблона. Специализация шаблона. Перегрузка шаблонов функции. Шаблоны класса. Правила описания шаблонов класса. Шаблоны как параметры шаблона класса
13.	Библиотека шаблонов	Стандартная библиотека шаблонов. Итераторы. Категории итераторов. Алгоритмы. Категории алгоритмов. Функции-предикаты: унарные и бинарные предикаты. Стандартные функциональные объекты. Свойства функциональных объектов. Собственные функциональные объекты.
14.	Потоковые классы. Файлы	Поток. Стандартные, файловые и строковые потоки. Использование файловых потоков для работы с файлами. Текстовые и бинарные файлы. Режимы открытия файла. Операции с файлом: создание, открытие, закрытие файла, чтение или запись. Функции для работы с файлами.
15.	Строки.	Конструкторы. Операции над строками. Доступ к элементу строки. Функции для работы со строками: присваивание и добавление частей строк, получение характеристик строк, преобразование строк, поиск подстрок, сравнение частей строк.
16.	Последовательные и ассоциативные контейнеры	Последовательные контейнеры. Адаптеры последовательных контейнеров. Базовые методы и характерные операции контейнеров. Ассоциативные

		контейнеры. Общие методы контейнеров.
17.	Коллекции	Индексируемая коллекция. Класс Array. Одномерные и многомерные массивы: объявление и инициализация. Класс List. Создание списка. Добавление и удаление элементов. Статические методы поиска и сортировки в индексируемых коллекциях. Обход коллекции с помощью оператора foreach. Коллекции пар "ключ-значение". Класс Dictionary. Создание словаря. Добавление пары ключ-значение. Обход элементов словаря. Поиск элементов по ключу. Строки как коллекции. Класс String. Неизменяемость строки. Основные статические и динамические методы класса String. Операции над строками.
18.	Обработка исключительных ситуаций	Исключение. Подсистема обработки исключений. Контролируемый блок кода. Обработчики исключений. Блок завершения. Генерация исключения. Перехват исключений. Класс Exception. Определение собственных классов исключений. Иерархия исключений.
19.	Работа с файловой системой	Класс Stream. Работа с текстовыми и бинарными файлами с помощью потоков: классы StreamReader и StreamWriter, BinaryReader и BinaryWriter. Сериализация. Кодировки текстовых файлов. Проверка наличия каталога и файла.
20.	Классы. Отношения вложенности	Клиенты и поставщики. Варианты реализации отношения "клиент-поставщик": создание объекта класса поставщика, вызов статических полей или методов класса поставщика. Отношения между клиентами и поставщиками.
21.	Методы класса с особым синтаксисом	Свойства. Определение свойства. Синтаксис автосвойств. Управление доступом для методов set и get. Вычисляемые свойства. Индексаторы. Объявления индексаторов. Сценарии использования индексаторов.
22.	Интерфейсы	Множественное наследование. Интерфейсы. Наследование интерфейсов. Отличие абстрактных классов и интерфейсов. Неявная реализация интерфейсов. Явная реализация: обертывание и кастинг. Коллизия имен интерфейсов. Склеивание и переименование методов. Встроенные интерфейсы.
23.	Делегаты	Делегаты. Свойства делегата. Объявление делегата. Экземпляр делегата. Одноадресный и многоадресный делегат. Механизм работы с делегатом. Использование лямбда-выражений.
24.	События	События. Свойства событий. Издатели и подписчики. Определение и вызов событий. Обработчики события. Подписка на события. Подписка на события с помощью анонимной функции. Отмена подписки на события.
25.	Построение приложений с графическим интерфейсом	Формы. Элементы управления. Группы элементов управления. Класс Control: свойства и методы. События класса Control. Параметры обработчика события. Программное управление интерфейсом. Группирование элементов. Главное меню. Контекстное меню. Панель

		инструментов. Строка состояния
26.	Диалоговые окна	Диалоговое окно MessageBox. Перечисления MessageBoxButtons и MessageBoxIcon. Идентификация нажатой кнопки: перечисление DialogResult. Стандартные диалоговые окна: OpenFileDialog, SaveFileDialog, FolderBrowserDialog, ColorDialog.
27.	Источники данных. Привязка к источнику данных	Источники данных. Простая привязка. Сложная привязка. Свойство DataSource. Компонент BindingSource. Режим визуальной привязки. Обновление привязки и перерисовка элемента управления. BindingNavigator.
28.	Взаимодействие форм приложения	Родительские и дочерние формы. Обмен данными между формами приложения. Стандартные диалоговые окна. Всплывающее окно. Управление формами в Main. Немодальные окна. Приложения с интерфейсом MDI.
29.	Способы табличного отображения данных	Элемент управления DataGridView. Коллекции объектов таблицы: столбцы, строки, ячейки. Способы загрузки данных в DataGridView: привязка к источнику данных, программным способом.
30.	Способы подключения к базам данных	Библиотеки для работы с базами данных. Поставщики данных. Строка подключения. Последовательность операций для доступа к данным. Представление данных с помощью DataGridView. Операции CRUD. Создание моделей.
31.	Язык интегрированных запросов LINQ и элементы функционального программирования	Механизм доступа к данным LINQ. LINQ при работе с массивами. SQL-стиль использования LINQ. Использование LINQ через методы. Использование LINQ для доступа к XML. Использование LINQ для работы с базами данных
32.	Многопоточность. Параллельное программирование	Параллельная программа. Класс Thread. Передача параметров в поток. Конкурентный доступ. Пул потоков. Ключевые слова async/await. Асинхронное программирование. Асинхронные методы. Типы возвращаемых значений

6.2.3. Содержание самостоятельной работы

№ п/п	Наименование темы (раздела) дисциплины	Содержание самостоятельной работы
1.	Алгоритмы. Способы представления алгоритмов	Алгоритм. Свойства алгоритмов. Основные этапы разработки алгоритмов: постановка задачи, построение математической модели, разработка алгоритма решения задачи, проверка правильности и оценка сложности алгоритма. Формы и способы представления алгоритмов. Базовые алгоритмические конструкции. Вспомогательные алгоритмы. Алгоритмически неразрешимые проблемы.
2.	Базовые элементы языка программирования	Классификация языков программирования. Синтаксис и семантика языка программирования. Формальные методы описания синтаксиса. Алфавит. Идентификаторы. Ключевые слова. Типы данных.

		Классификация типов. Конверсия типов. Переменные. Константы. Операции. Выражения. Классификация и приоритет операций. Операторы. Комментарии.
3.	Средства разработки программ	Интегрированная среда разработки. Архитектурные типы. Основные компоненты: поиск и навигация, редактирование и рефакторинг, отладка и тестирование. Система управления версиями.
4.	Базовые конструкции структурного программирования	Теорема Бёма — Якопини. Структура программы. Комментарии. Стандартный ввод/вывод. Конструкция следования. Стандартные функции языка. Организация ветвлений: бинарное и множественное ветвление. Тернарная операция. Множественный выбор. Организация циклов. Вложенные циклы. Принудительное завершение итерации и цикла.
5.	Функции	Описание и определение функций. Вызов функции. Возвращаемое значение функции. Параметры функции: позиционные, по умолчанию, именованные. Функция с переменным числом параметров. Передача параметров по значению и по адресу. -функции. Использование глобальных переменных. Рекурсивная функция. Перегрузка функций. Правила описания перегруженных функций. Схема алгоритма поиска функции.
6.	Массивы данных. Алгоритмы сортировки данных	Одномерные и многомерные массивы. Определение и инициализация. Доступ к элементу массива. Пакетная обработка массива. Линейный и бинарный поиск в массиве. Модификация массивов. Передача массивов в функцию. Массивы символов. Сортировка. Классификация сортировок. Основные виды сортировок: пузырьковая, выбором, шейкерная, расческой. Эффективные сортировки: быстрая, слиянием.
7.	Указатели и ссылки. Динамическое распределение памяти	Указатели. Виды указателей. Способы инициализации указателей. Операции с указателями. Арифметические операции с указателями. Ссылки. Правила инициализации ссылок. Виды ссылок. Ссылки на массив и на функцию. Динамически распределяемая память. Выделение динамической памяти: оператор new. Освобождение динамической памяти: оператор delete. Указатели и динамическая память. Массивы в динамической памяти. Передача динамических массивов в функции. Отслеживание утечек памяти.
8.	Типы данных, определяемые пользователем	Перечисляемый тип. Структура. Инициализация структуры. Элементы структуры. Доступ к полям структуры. Битовые поля. Объединения. Ограничения накладываемые на объединения.
9.	ООП. Классы и объекты.	ООП. Классы. Описание класса. Члены класса. Управление доступом к членам класса. Экземпляры класса. Доступ к элементам экземпляра. Конструкторы. Указатель this. Статические элементы класса. Расширение интерфейса класса. Деструкторы

10.	Отношение наследования	Наследование. Базовые и производные классы. Ключи доступа. Одиночное наследование. Прямое и косвенное наследование. Правила наследования членов класса. Замещение метода базового класса. Иерархия классов.
11.	Механизмы полиморфизма	Полиморфное поведение. Раннее и позднее связывание. Виртуальные методы. Правила использования виртуальных методов. Абстрактные классы. Перегрузка операций.
12.	Шаблоны функций	Шаблоны функций. Шаблоны функций. Шаблоны параметров. Инстанцирование шаблона. Специализация шаблона. Перегрузка шаблонов функции. Шаблоны класса. Правила описания шаблонов класса. Шаблоны как параметры шаблона класса
13.	Библиотека шаблонов	Стандартная библиотека шаблонов. Итераторы. Категории итераторов. Алгоритмы. Категории алгоритмов. Функции-предикаты: унарные и бинарные предикаты. Стандартные функциональные объекты. Свойства функциональных объектов. Собственные функциональные объекты.
14.	Потоковые классы. Файлы	Поток. Стандартные, файловые и строковые потоки. Использование файловых потоков для работы с файлами. Текстовые и бинарные файлы. Режимы открытия файла. Операции с файлом: создание, открытие, закрытие файла, чтение или запись. Функции для работы с файлами.
15.	Строки.	Конструкторы. Операции над строками. Доступ к элементу строки. Функции для работы со строками: присваивание и добавление частей строк, получение характеристик строк, преобразование строк, поиск подстроки, сравнение частей строк.
16.	Последовательные и ассоциативные контейнеры	Последовательные контейнеры. Адаптеры последовательных контейнеров. Базовые методы и характерные операции контейнеров. Ассоциативные контейнеры. Общие методы контейнеров.
17.	Коллекции	Индексируемая коллекция. Класс Array. Одномерные и многомерные массивы: объявление и инициализация. Класс List. Создание списка. Добавление и удаление элементов. Статические методы поиска и сортировки в индексируемых коллекциях. Обход коллекции с помощью оператора foreach. Коллекции пар "ключ-значение". Класс Dictionary. Создание словаря. Добавление пары ключ-значение. Обход элементов словаря. Поиск элементов по ключу. Строки как коллекции. Класс String. Неизменяемость строки. Основные статические и динамические методы класса String. Операции над строками.
18.	Обработка исключительных ситуаций	Исключение. Подсистема обработки исключений. Контролируемый блок кода. Обработчики исключений. Блок завершения. Генерация исключения. Перехват исключений. Класс Exception. Определение собственных классов исключений. Иерархия

		исключений.
19.	Работа с файловой системой	Класс Stream. Работа с текстовыми и бинарными файлами с помощью потоков: классы StreamReader и StreamWriter, BinaryReader и BinaryWriter. Сериализация. Кодировки текстовых файлов. Проверка наличия каталога и файла.
20.	Классы. Отношения вложенности	Клиенты и поставщики. Варианты реализации отношения "клиент-поставщик": создание объекта класса поставщика, вызов статических полей или методов класса поставщика. Отношения между клиентами и поставщиками.
21.	Методы класса с особым синтаксисом	Свойства. Определение свойства. Синтаксис автосвойств. Управление доступом для методов set и get. Вычисляемые свойства. Индексаторы. Объявления индексаторов. Сценарии использования индексаторов.
22.	Интерфейсы	Множественное наследование. Интерфейсы. Наследование интерфейсов. Отличие абстрактных классов и интерфейсов. Неявная реализация интерфейсов. Явная реализация: обертывание и кастинг. Коллизия имен интерфейсов. Склеивание и переименование методов. Встроенные интерфейсы.
23.	Делегаты	Делегаты. Свойства делегата. Объявление делегата. Экземпляр делегата. Одноадресный и многоадресный делегат. Механизм работы с делегатом. Использование лямбда-выражений.
24.	События	События. Свойства событий. Издатели и подписчики. Определение и вызов событий. Обработчики события. Подписка на события. Подписка на события с помощью анонимной функции. Отмена подписки на события.
25.	Построение приложений с графическим интерфейсом	Формы. Элементы управления. Группы элементов управления. Класс Control: свойства и методы. События класса Control. Параметры обработчика события. Программное управление интерфейсом. Группирование элементов. Главное меню. Контекстное меню. Панель инструментов. Строка состояния
26.	Диалоговые окна	Диалоговое окно MessageBox. Перечисления MessageBoxButtons и MessageBoxIcon. Идентификация нажатой кнопки: перечисление DialogResult. Стандартные диалоговые окна: OpenFileDialog, SaveFileDialog, FolderBrowserDialog, ColorDialog.
27.	Источники данных. Привязка к источнику данных	Источники данных. Простая привязка. Сложная привязка. Свойство DataSource. Компонент BindingSource. Режим визуальной привязки. Обновление привязки и перерисовка элемента управления. BindingNavigator.
28.	Взаимодействие форм приложения	Родительские и дочерние формы. Обмен данными между формами приложения. Стандартные диалоговые окна. Всплывающее окно. Управление формами в Main. Немодальные окна. Приложения с интерфейсом MDI.

29.	Способы табличного отображения данных	Элемент управления DataGridView. Коллекции объектов таблицы: столбцы, строки, ячейки. Способы загрузки данных в DataGridView: привязка к источнику данных, программным способом.
30.	Способы подключения к базам данных	Библиотеки для работы с базами данных. Поставщики данных. Строка подключения. Последовательность операций для доступа к данным. Представление данных с помощью DataGridView. Операции CRUD. Создание моделей.
31.	Язык интегрированных запросов LINQ и элементы функционального программирования	Механизм доступа к данным LINQ. LINQ при работе с массивами. SQL-стиль использования LINQ. Использование LINQ через методы. Использование LINQ для доступа к XML. Использование LINQ для работы с базами данных
32.	Многопоточность. Параллельное программирование	Параллельная программа. Класс Thread. Передача параметров в поток. Конкурентный доступ. Пул потоков. Ключевые слова async/await. Асинхронное программирование. Асинхронные методы. Типы возвращаемых значений

7. Текущий контроль по дисциплине (модулю) в рамках учебных занятий

В рамках текущего контроля преподаватель самостоятельно может проводить следующие мероприятия:

№ п/п	Контролируемые разделы (темы)	Наименование оценочного средства
1.	Алгоритмы. Способы представления алгоритмов	Опрос, практическое задание, тестирование
2.	Базовые элементы языка программирования	Опрос, практическое задание, тестирование
3.	Средства разработки программ	Опрос, практическое задание, тестирование
4.	Базовые конструкции структурного программирования	Опрос, практическое задание, тестирование
5.	Функции	Опрос, практическое задание, тестирование
6.	Массивы данных. Алгоритмы сортировки данных	Опрос, практическое задание, тестирование
7.	Указатели и ссылки. Динамическое распределение памяти	Опрос, практическое задание, тестирование
8.	Типы данных, определяемые пользователем	Опрос, практическое задание, тестирование
9.	ООП. Классы и объекты.	Опрос, практическое задание, тестирование
10.	Отношение наследования	Опрос, практическое задание, тестирование
11.	Механизмы полиморфизма	Опрос, практическое задание, тестирование

12.	Шаблоны функций	Опрос, практическое задание, тестирование
13.	Библиотека шаблонов	Опрос, практическое задание, тестирование
14.	Потоковые классы. Файлы	Опрос, практическое задание, тестирование
15.	Строки.	Опрос, практическое задание, тестирование
16.	Последовательные и ассоциативные контейнеры	Опрос, практическое задание, тестирование
17.	Коллекции	Опрос, практическое задание, тестирование
18.	Обработка исключительных ситуаций	Опрос, практическое задание, тестирование
19.	Работа с файловой системой	Опрос, практическое задание, тестирование
20.	Классы. Отношения вложенности	Опрос, практическое задание, тестирование
21.	Методы класса с особым синтаксисом	Опрос, практическое задание, тестирование
22.	Интерфейсы	Опрос, практическое задание, тестирование
23.	Делегаты	Опрос, практическое задание, тестирование
24.	События	Опрос, практическое задание, тестирование
25.	Построение приложений с графическим интерфейсом	Опрос, практическое задание, тестирование
26.	Диалоговые окна	Опрос, практическое задание, тестирование
27.	Источники данных. Привязка к источнику данных	Опрос, практическое задание, тестирование
28.	Взаимодействие форм приложения	Опрос, практическое задание, тестирование
29.	Способы табличного отображения данных	Опрос, практическое задание, тестирование
30.	Способы подключения к базам данных	Опрос, практическое задание, тестирование
31.	Язык интегрированных запросов LINQ и элементы функционального программирования	Опрос, практическое задание, тестирование
32.	Многопоточность. Параллельное программирование	Опрос, практическое задание, тестирование

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля)

8.1. Основная учебная литература

1. Вязовик, Н. А. Программирование на Java : учебное пособие / Н. А. Вязовик. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 601 с. — ISBN 978-5-4497-0852-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/146383.html>
2. Маркин, А. В. Web-программирование : учебник / А. В. Маркин. — 2-е изд. — Москва : Ай Пи Ар Медиа, 2024. — 286 с. — ISBN 978-5-4497-3244-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/141273.html>
3. Котова, О. В. Алгоритмизация и программирование : учебное пособие для студентов направления подготовки 09.03.02 Информационные системы и технологии / О. В. Котова, Ю. В. Скидан. — Макеевка : Донбасская национальная академия строительства и архитектуры, ЭБС АСВ, 2024. — 118 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/141647.html>

8.2. Дополнительная учебная литература:

1. Разработка Windows-приложений в среде программирования Visual Studio.Net : учебно-методическое пособие по дисциплине Информатика и программирование / составители Ю. А. Воронцов, А. Г. Ерохин. — Москва : Московский технический университет связи и информатики, 2016. — 20 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/61536.html>
2. Флloyd, К. С. Введение в программирование на PHP5 : учебное пособие / К. С. Флloyd. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 280 с. — ISBN 978-5-4497-0886-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/146333.html>

9. Перечень ресурсов информационно-телекоммуникационной сети "Интернет" (далее - сеть "Интернет"), необходимых для освоения дисциплины (модуля)

1. Федеральный портал «Российское образование». <http://www.edu.ru>
2. Научная электронная библиотека eLIBRARY.RU (ресурсы открытого доступа) <https://www.elibrary.ru/defaultx.asp>
3. <https://www.rsl.ru/> Российская Государственная Библиотека (ресурсы открытого доступа)

10. Методические указания для обучающихся по освоению дисциплины (модуля)

Успешное освоение данного курса базируется на рациональном сочетании нескольких видов учебной деятельности – лекций, семинарских занятий, самостоятельной работы. При этом самостоятельную работу следует рассматривать одним из главных звеньев полноценного высшего образования, на которую отводится значительная часть учебного времени.

Самостоятельная работа студентов складывается из следующих составляющих:

- работа с основной и дополнительной литературой, с материалами интернета и конспектами лекций;
- внеаудиторная подготовка к контрольным работам, выполнение докладов, рефератов и курсовых работ;
- выполнение самостоятельных практических работ;
- подготовка к экзаменам (зачетам) непосредственно перед ними.

Для правильной организации работы необходимо учитывать порядок изучения разделов курса, находящихся в строгой логической последовательности. Поэтому хорошее усвоение одной части дисциплины является предпосылкой для успешного перехода к следующей. Задания, проблемные вопросы, предложенные для изучения дисциплины, в том числе и для самостоятельного выполнения, носят междисциплинарный характер и базируются, прежде всего, на причинно-следственных связях между компонентами окружающего нас мира. В течение семестра, необходимо подготовить рефераты (проекты) с использованием рекомендуемой основной и дополнительной литературы и сдать рефераты для проверки преподавателю. Важным составляющим в изучении данного курса является решение ситуационных задач и работа над проблемно-аналитическими заданиями, что предполагает знание соответствующей научной терминологии и т.д.

Для лучшего запоминания материала целесообразно использовать индивидуальные особенности и разные виды памяти: зрительную, слуховую, ассоциативную. Успешному запоминанию также способствует приведение ярких свидетельств и наглядных примеров. Учебный материал должен постоянно повторяться и закрепляться.

При выполнении докладов, творческих, информационных, исследовательских проектов особое внимание следует обращать на подбор источников информации и методику работы с ними.

Для успешной сдачи экзамена (зачета) рекомендуется соблюдать следующие правила:

1. Подготовка к экзамену (зачету) должна проводиться систематически, в течение всего семестра.
2. Интенсивная подготовка должна начаться не позднее, чем за месяц до экзамена.
3. Время непосредственно перед экзаменом (зачетом) лучше использовать таким образом, чтобы оставить последний день свободным для повторения курса в целом, для систематизации материала и доработки отдельных вопросов.

На экзамене высокую оценку получают студенты, использующие данные, полученные в процессе выполнения самостоятельных работ, а также использующие собственные выводы на основе изученного материала.

Учитывая значительный объем теоретического материала, студентам рекомендуется регулярное посещение и подробное конспектирование лекций.

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

1. Терминальный сервер, предоставляющий к нему доступ клиентам на базе Windows Server 2016
2. Семейство ОС Microsoft Windows
3. Libre Office свободно распространяемый офисный пакет с открытым исходным кодом
4. Информационно-справочная система: Система КонсультантПлюс (Информационный комплекс)
5. Информационно-правовое обеспечение Гарант: Электронный периодический справочник «Система ГАРАНТ» (ЭПС «Система ГАРАНТ»)

Перечень используемого программного обеспечения указан в п.12 данной рабочей

программы дисциплины.

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

12.1. Учебная аудитория для проведения учебных занятий, предусмотренных программой, оснащенная оборудованием и техническими средствами обучения.

Специализированная мебель:

Комплект учебной мебели (стол, стул) по количеству обучающихся; доска (маркерная), комплект мебели для преподавателя.

Технические средства обучения:

Компьютер в сборе для преподавателя; компьютеры в сборе для обучающихся; колонки; проектор, экран.

Перечень лицензионного программного обеспечения, в том числе отечественного производства:

Операционные системы семейства Windows, КонсультантПлюс веб версия, Гарант веб версия, антивирус Kaspersky Endpoint Security.

Перечень свободно распространяемого программного обеспечения:

Яндекс Браузер, LibreOffice, МТС Линк, Notepad++. Pinta, Gimp, AnyLogic, Inkscape, OpenShot. FreeCAD, LibreCAD, Jamovi, Visual Studio, Unity.

Подключение к сети «Интернет» и обеспечение доступа в электронную информационно-образовательную среду ММУ.

12.2. Помещение для самостоятельной работы обучающихся.

Специализированная мебель:

Комплект учебной мебели (стол, стул) по количеству обучающихся; доска (маркерная), комплект мебели для преподавателя.

Технические средства обучения:

Компьютер в сборе для преподавателя; компьютеры в сборе для обучающихся; колонки; проектор, экран.

Перечень лицензионного программного обеспечения, в том числе отечественного производства:

Операционные системы семейства Windows, КонсультантПлюс веб версия, Гарант веб версия, антивирус Kaspersky Endpoint Security.

Перечень свободно распространяемого программного обеспечения:

Яндекс Браузер, LibreOffice, МТС Линк, Notepad++. Pinta, Gimp, AnyLogic, Inkscape, OpenShot. FreeCAD, LibreCAD, Jamovi, Visual Studio, Unity.

Помещение для самостоятельной работы обучающихся оснащено компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду ММУ.

13. Образовательные технологии, используемые при освоении дисциплины

Для освоения дисциплины используются как традиционные формы занятий – лекции (типы лекций – установочная, вводная, текущая, заключительная, обзорная; виды лекций – проблемная, визуальная, лекция конференция, лекция консультация); и семинарские (практические) занятия, так и активные и интерактивные формы занятий - деловые и ролевые игры, решение ситуационных задач и разбор конкретных ситуаций.

На учебных занятиях используются технические средства обучения мультимедийной аудитории: компьютер, монитор, колонки, настенный экран, проектор, микрофон, пакет программ Microsoft Office для демонстрации презентаций и медиафайлов, видеопроектор для демонстрации слайдов, видеосюжетов и др. Тестирование обучаемых может осуществляться с использованием компьютерного оборудования университета.

13.1. В освоении учебной дисциплины используются следующие традиционные образовательные технологии:

- чтение проблемно-информационных лекций с использованием доски и видеоматериалов;
- семинарские занятия для обсуждения, дискуссий и обмена мнениями;
- контрольные опросы;
- консультации;
- самостоятельная работа студентов с учебной литературой и первоисточниками;
- подготовка и обсуждение рефератов (проектов), презентаций (научно-исследовательская работа);
- тестирование по основным темам дисциплины.

13.2. Активные и интерактивные методы и формы обучения

Из перечня видов: («мозговой штурм», анализ НПА, анализ проблемных ситуаций, анализ конкретных ситуаций, инциденты, имитация коллективной профессиональной деятельности, разыгрывание ролей, творческая работа, связанная с освоением дисциплины, ролевая игра, круглый стол, диспут, беседа, дискуссия, мини-конференция и др.) используются следующие:

- диспут
- анализ проблемных, творческих заданий, ситуационных задач
- ролевая игра;
- круглый стол;
- мини-конференция
- дискуссия
- беседа.

13.3 Особенности обучения инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ)

При организации обучения по дисциплине учитываются особенности организации взаимодействия с инвалидами и лицами с ограниченными возможностями здоровья (далее – инвалиды и лица с ОВЗ) с целью обеспечения их прав. При обучении учитываются особенности их психофизического развития, индивидуальные возможности и при необходимости обеспечивается коррекция нарушений развития и социальная адаптация указанных лиц.

Выбор методов обучения определяется содержанием обучения, уровнем методического и материально-технического обеспечения, особенностями восприятия учебной информации студентами-инвалидами и студентами с ограниченными возможностями здоровья и т.д. В образовательном процессе используются социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе.

При обучении лиц с ограниченными возможностями здоровья электронное обучение и дистанционные образовательные технологии предусматривают возможность приема-передачи информации в доступных для них формах.

Обучающиеся из числа лиц с ограниченными возможностями здоровья обеспечены печатными и электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

**Автономная некоммерческая организация высшего образования
«МОСКОВСКИЙ МЕЖДУНАРОДНЫЙ УНИВЕРСИТЕТ»**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ДИСЦИПЛИНЕ**

Алгоритмизация и программирование

<i>Направление подготовки</i>	Прикладная информатика
<i>Код</i>	09.03.03
<i>Направленность (профиль)</i>	Прикладная информатика в бизнесе
<i>Квалификация выпускника</i>	бакалавр

1. Перечень кодов компетенций, формируемых дисциплиной в процессе освоения образовательной программы

Группа компетенций	Категория компетенций	Код
Общепрофессиональные	-	ОПК-2
Общепрофессиональные	-	ОПК-3
Общепрофессиональные	-	ОПК-4
Общепрофессиональные	-	ОПК-7

2. Компетенции и индикаторы их достижения

Код компетенции	Формулировка компетенции	Индикаторы достижения компетенции
ОПК-2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1 Знает принципы работы современных информационных технологий и программных средств, в том числе отечественного производства при решении задач профессиональной деятельности ОПК-2.2 Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности ОПК-2.3 Владеет навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности
ОПК-3	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	ОПК-3.1 Знает принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности ОПК-3.2 Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности ОПК-3.3 Владеет навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций,

		и библиографии по научно-исследовательской работе с учетом требований информационной безопасности
ОПК-4	Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	ОПК-4.1 Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.2 Умеет применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.3 Владеет навыками составления технической документации на различных этапах жизненного цикла информационной системы
ОПК-7	Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1 Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий ОПК-7.2 Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ ОПК-7.3 Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач

3. Описание планируемых результатов обучения по дисциплине и критериев оценки результатов обучения по дисциплине

3.1. Описание планируемых результатов обучения по дисциплине

Планируемые результаты обучения по дисциплине представлены дескрипторами (знания, умения, навыки)

Дескрипторы по дисциплине	Знать	Уметь	Владеть
Код компетенции	ОПК-2		
	- особенности и ограничения применимости инструментов для известного контекста	- обоснованно выбрать необходимые технологии для решения поставленной задачи	- организации межпрограммного взаимодействия для решения прикладных задач конечного пользователя; -

			систематизации программного обеспечения
Код компетенции	ОПК-3		
	- классификацию и критерии классификации информационных технологий; - характеристики базовых информационных процессов сбора, передачи, обработки, хранения и представления информации, а также средства реализации базовых информационных процессов	- работать с информацией в глобальных компьютерных сетях и корпоративных информационных системах; - использовать системы поиска профессиональной информации в глобальных сетях	- работы в локальных и глобальных сетях при решении научных и исследовательских задач; - использования программных средств обеспечения безопасности данных на автономном ПК и в интерактивной среде
Код компетенции	ОПК-4		
	- основные принципы процесса разработки программного обеспечения; - методы и средства разработки программной документации	- формализовать прикладную задачу, выбирать для неё подходящие структуры данных и алгоритмы обработки; - пользоваться основными методологиями процессов разработки программного обеспечения и документации	- навыками применения методов и средств разработки программной документации
Код компетенции	ОПК-7		
	- принципы объектно-ориентированного программирования; - языки и среды программирования; - основные этапы разработки программного обеспечения	- программировать алгоритмы, используя средства языков программирования; - создавать программу по разработанному алгоритму как отдельный модуль	- проведения тестирования программного модуля по определенному сценарию

3.2. Критерии оценки результатов обучения по дисциплине

Шкала оценив ания	Индикатор ы достижени я	Показатели оценивания результатов обучения
ОТЛИЧНО/ЗАЧТЕНО	Знает:	- студент глубоко и всесторонне усвоил материал, уверенно, логично, последовательно и грамотно его излагает, опираясь на знания основной и дополнительной литературы, - на основе системных научных знаний делает квалифицированные выводы и обобщения, свободно оперирует категориями и понятиями.
	Умеет:	- студент умеет самостоятельно и правильно решать учебно-профессиональные задачи или задания, уверенно, логично, последовательно и аргументировано излагать свое решение, используя научные понятия, ссылаясь на нормативную базу.
	Владеет:	- студент владеет рациональными методами (с использованием рациональных методик) решения сложных профессиональных задач, представленных деловыми играми, кейсами и т.д.; - При решении продемонстрировал навыки - выделения главного, - связкой теоретических положений с требованиями руководящих документов, - изложения мыслей в логической последовательности, - самостоятельного анализа факты, событий, явлений, процессов в их взаимосвязи и диалектическом развитии.
ХОРОШО/ЗАЧТЕНО	Знает:	- студент твердо усвоил материал, достаточно грамотно его излагает, опираясь на знания основной и дополнительной литературы, - затрудняется в формулировании квалифицированных выводов и обобщений, оперирует категориями и понятиями, но не всегда правильно их верифицирует.
	Умеет:	- студент умеет самостоятельно и в основном правильно решать учебно-профессиональные задачи или задания, уверенно, логично, последовательно и аргументировано излагать свое решение, не в полной мере используя научные понятия и ссылки на нормативную базу.
	Владеет:	- студент в целом владеет рациональными методами решения сложных профессиональных задач, представленных деловыми играми, кейсами и т.д.; - При решении смог продемонстрировать достаточность, но не глубинность навыков - выделения главного, - изложения мыслей в логической последовательности. - связки теоретических положений с требованиями руководящих документов, - самостоятельного анализа факты, событий, явлений, процессов в их взаимосвязи и диалектическом развитии.

УДОВЛЕТВОРИТЕЛЬНО/ЗАЧТЕНО	Знает:	- студент ориентируется в материале, однако затрудняется в его изложении; - показывает недостаточность знаний основной и дополнительной литературы; - слабо аргументирует научные положения; - практически не способен сформулировать выводы и обобщения; - частично владеет системой понятий.
	Умеет:	- студент в основном умеет решить учебно-профессиональную задачу или задание, но допускает ошибки, слабо аргументирует свое решение, недостаточно использует научные понятия и руководящие документы.
	Владеет:	- студент владеет некоторыми рациональными методами решения сложных профессиональных задач, представленных деловыми играми, кейсами и т.д.; При решении продемонстрировал недостаточность навыков - выделения главного, - изложения мыслей в логической последовательности. - связки теоретических положений с требованиями руководящих документов, - самостоятельного анализа факты, событий, явлений, процессов в их взаимосвязи и диалектическом развитии.
Компетенция не достигнута		
НЕУДОВЛЕТВОРИТЕЛЬНО/ НЕ ЗАЧТЕНО	Знает:	- студент не усвоил значительной части материала; - не может аргументировать научные положения; - не формулирует квалифицированных выводов и обобщений; - не владеет системой понятий.
	Умеет:	студент не показал умение решать учебно-профессиональную задачу или задание.
	Владеет:	не выполнены требования, предъявляемые к навыкам, оцениваемым “удовлетворительно”.

4. Типовые контрольные задания и/или иные материалы для проведения промежуточной аттестации, необходимые для оценки достижения компетенции

ОПК-2

Типовые тесты

Вопрос №1. Какие из перечисленных структур относятся к алгоритмам?

Тип ответа: Многие из многих

Варианты ответов:

1. разветвляющаяся
2. линейная
3. объектно-ориентированная
4. циклическая
5. иерархическая

Вопрос №2. Как найти квадратный корень из числа x

Варианты ответов:

1. Sqrt(x)
2. Summ.Koren(x)
3. Summ.Koren(x)
4. Math.Sqrt(x)

Вопрос №3. Что делает оператор «%»

Варианты ответов:

1. Возвращает процент от суммы
2. Возвращает остаток от деления
3. Возвращает тригонометрическую функцию
4. Ни чего из выше перечисленного

Вопрос №4. Что сделает программа, выполнив следующий код: Console.WriteLine(«Hello, World!»);

Варианты ответов:

1. Напишет на новой строке Hello, World!
2. Напишет Hello, World!
3. Удалит все значения с Hello, World!
4. Вырежет слово Hello, World! из всего текста

Вопрос №5. Как сделать инкрементацию числа

Варианты ответов:

1. ++
2. —
3. %%
4. !=

Практические задания

Лабораторная работа

Задача 1.

Решение вспомогательных задач для усвоения теоретических основ ИБ. Диагностика и настройка персонального компьютера

Задания:

1. Настройка средств ввода-вывода операционной системы Windows.
2. Настройка элементов управления Windows.
3. Резервное копирование данных.
4. Проверка жесткого диска.

Задача 2.

Формирование требований к системам защиты информации в виде профилей защиты в рамках идеологии «Общих критериев».

Задания

1. Сформулируйте интересы государства, общества и личности в информационной сфере
2. Чем определяется ценность информации для владельца?
3. В чем заключается комплексное обеспечение ИБ РФ?
4. Каковы основные методы и средства защиты процессов переработки информации в защищенных КС?
5. Назовите основные принципы процессов переработки информации.

6. Назовите основные виды угроз.

Задача 3.

Ознакомление со штатными средствами ОС по обеспечению информационной безопасности на примере WINDOWS NT(XP).

Задания:

1. Создание и установка прав доступа пользователей.
2. Установка прав доступа к объектам

Задача 3.1. Защита документа в Microsoft Word. Восстановление текста, поврежденного документ. Изучить возможности ограничения изменений в документе:

Задания: 1. Установить в документе пароль для открытия документа, руководствуясь правилами, описанными в работе.

2. Установить в документе пароль разрешения записи.
3. Установить режим: Рекомендовать доступ только для чтения.
4. Проверить не содержит ли документ скрытых данных.
5. Изменить интервал времени автоматического сохранения документов.
6. Установить режим сохранения резервной копии документа.

Задача 3.2. Защита документа в Microsoft Excel. Изучить возможности ограничения просмотра и изменения пользователями данных в электронных таблицах.

Задания:

1. Установить пароль для открытия книги.
2. Установить пароль для разрешения записи.
3. Установить защиту ячеек.
4. Открыть несколько книг, скрыть одну из них.
5. Отобразить скрытую книгу
6. Скрыть лист.
7. Скрыть изображение столбца.
8. Отобразить скрытый столбец.
9. Скрыть изображение строки.
10. Отобразить скрытую строку.

Задача 3.3. Работа с реестром ОС. Изучить основные принципы работы с реестром, освоить редактор реестра, научиться создавать резервные копии как реестра целиком, так и его отдельных ключей

Задания:

1. Сохранить значение всей ветви HKEY_CLASSES_ROOT.
2. Включ. HKEY_CLASSES_ROOT найти ветвь lnkfile. Одним из ее параметров является IsShortcut. Удалите его. Аналогичную процедуру повторите с ветвью piffile. Перезагрузите компьютер. Обратите внимание, что исчезли все стрелки с ярлыков программ.
3. Восстановить значение ветви HKEY_CLASSES_ROOT.
4. Создать резервную копию файла реестра.
5. Откройте ключ реестра HKEY_CURRENT_USER, а затем его подключите \ControlPanel\Desktop. Добавьте к открытому ключу новое строковое значение с именем MenuShowDelay. Дважды щелкните мышью, указав на это значение и введите число 1. Затем перезагрузите систему. Теперь меню, запрашиваемые с панели задач, будут появлять-ся гораздо быстрее.
6. Восстановите реестр с резервной копии.
7. С помощью программы Microsoft Backup создать копию реестра, а затем по этой копии восстановить реестр.
8. Перезагрузите систему и убедитесь, что она функционирует нормально.

Задача 4.

Использование архиваторов для защиты информации.

Задания:

1. Выделить группы архивируемых файлов в WinRAR.
2. Создать различных типов архивов в WinRAR и работа с ними.
3. Выполнить шифрование информации в WinRAR.

Задача 5.

Изучение основных принципов уничтожения и восстановления информации на магнитных дисках, знакомство с используемыми утилитами, входящими в пакет NortonUtilities.

Задания:

1. Написать командный файл, при запуске которого произойдет затирание файлов с расширением ВАК на жестком диске С. Использовать программу WipeInfo.
2. Удалить на жестком диске несколько файлов, а затем попытаться с помощью программы UnErase восстановить их. Поэкспериментировать для случая, когда файлы удаляются вместе с подкаталогами, содержащими их.
3. Проверить жесткий диск и дискету на наличие сбоев с помощью программы NortonDiskDoctor. Создать ситуацию, когда на дискете могут появиться потерянные кластеры и исправить их.

Задача 6.

Защита информации с помощью антивирусных программных средств. Использование электронной цифровой подписи.

Задания:

1. Изучить настройки программы DoctorWeb.
2. Провести тестирование системных областей жесткого диска и нескольких подкаталогов.
3. Проверить дискету на наличие вирусов.

Задача 7.

Защита информации с помощью шифрования данных, программы PGP (создание электронной цифровой подписи). Освоение работы с механизмами шифрования данных и электронной подписи

Задания:

Шифрование информационных массивов методами битовых манипуляций, подстановки, перестановки, замены.

Задача 8.

Защита информации при работе с электронной почтой.

Задания:

1. Создание и настройка параметров безопасности электронного почтового ящика.
2. Прием и рассылка электронных писем.
3. Работа с электронной почтой с помощью почтового клиента MicrosoftOutlookExpress.

Задача 9.

Для последовательного поиска определите:

- максимальное и среднее число сравнений при поиске в неупорядоченном массиве из M элементов единственного значения q ;
- среднее число сравнений при поиске в упорядоченном массиве из M элементов единственного значения q , когда искомого значения в массиве нет.

Задания открытого типа

1. Алгоритм. Свойства алгоритмов.
2. Основные этапы разработки алгоритмов: постановка задачи, построение математической модели, разработка алгоритма решения задачи, проверка правильности и оценка сложности алгоритма.
3. Формы и способы представления алгоритмов.
4. Базовые алгоритмические конструкции. Вспомогательные алгоритмы.
5. Алгоритмически неразрешимые проблемы.
6. Классификация языков программирования.
7. Синтаксис и семантика языка программирования.
8. Формальные методы описания синтаксиса.
9. Алфавит. Идентификаторы. Ключевые слова.
10. Типы данных. Классификация типов. Конверсия типов.
11. Переменные. Константы.
12. Операции. Выражения. Классификация и приоритет операций.
13. Наследование. Базовые и производные классы.
14. Ключи доступа.
15. Одиночное наследование. Прямое и косвенное наследование.
16. Правила наследования членов класса.
17. Замещение метода базового класса.
18. Иерархия классов.
19. Полиморфное поведение.
20. Шаблоны функций.
21. Стандартная библиотека шаблонов.
22. Собственные функциональные объекты.
23. Поток. Стандартные, файловые и строковые потоки.
24. Режимы открытия файла. Операции с файлом: создание, открытие, закрытие файла, чтение или запись.
25. Конструкторы.

ОПК-3

Типовые тесты

Вопрос №1. Для чего нужны условные операторы

Варианты ответов:

1. Чтобы устанавливать условия пользователю
2. Для ветвления программы
3. Для оптимизации программы
4. Чтобы были

Вопрос №2. Укажите команды, в результате выполнения которых после вывода данных осуществляется переход на следующую строку

Тип ответа: Многие из многих

Варианты ответов:

1. `Cout << «Ocinka» << 12 << endl;`
2. `Cout << endl << «Ocinka» << 12;`
3. `Cout << «Ocinka» << 12 << «\n»;`
4. `Cout << «Ocinka \n» << 12;`

Вопрос №3. Обозначение оператора «ИЛИ»

Варианты ответов:

1. Or
2. !
3. !=
4. ||

Вопрос №4. Какие бывают циклы?

Варианты ответов:

1. Большие и маленькие
2. Цикл, Форич, Двойной цикл, Многократный
3. for, while, do-while, foreach
4. ref, out, static, root

Вопрос №5. Чему равен d, если `int a = 0; int b = a++; int c = 0`

Варианты ответов:

1. 3
2. True
3. False
4. 4

Практические задания Лабораторная работа

№1.

Установите адреса связи и указатели для удаления записи с ключом 85 из цепного каталога:

УС 11 УСП 10		
Адрес записи	Значение ключа	Адрес связи
10		0
11	16	14
12	37	13
13	85	16
14	19	15
15	33	12
16	91	0

№2.

Массив из 5 элементов целого типа заполнить при создании натуральным рядом чисел. Поменять местами 1-й и 5-й элемент. В 3-й элемент поместить сумму 1-го и 5-го.

Создать массив *m* плавающего типа из 10 элементов. Заполнить созданный массив тремя способами:

- 1) с консоли (клавиатуры) через приглашение на ввод данных
- 2) генератором случайных чисел в диапазоне (-100, +100)
- 3) по формуле: $m[i] = \sin(i) + \cos(i)$;

Создать два массива (вектора) одинаковой размерности, найти сумму и разность векторов.

Найти скалярное произведение двух векторов.

Создать вектор (массив) вещественных чисел из 20 элементов, заполнить случайными значениями в диапазоне (1, 10).

Найти модуль заданного вектора.

№3.

Массив целых чисел заполнить случайными значениями в диапазоне: $-10 + 10$. Для положительных элементов найти среднее значение.

Массив целых чисел из 10 элементов заполнить случайными значениями от 0 до 100. Вывести индекс 1-го элемента значение которого больше 50.

Матрицу размером 5×5 заполнить случайными значениями. Найти сумму элементов нулевого столбца.

Матрицу размером 5×5 заполнить случайными значениями. Найти сумму элементов побочной диагонали.

Матрицу размером 5×5 заполнить случайными значениями. Найти среднее значение элементов четвертого столбца.

Матрицу размером 5×5 заполнить случайными значениями. Найти сумму элементов главной диагонали.

№4.

Написать функцию, определяющую содержимое заданного бита целого числа.

Функции передаются, как параметры целое число и номер проверяемого бита.

Реализовать функцию для нахождения факториала числа: $f(n) = n!$

Написать функцию, реализующую заданную логическую операцию языка C++.

Написать функцию, выполняющую сортировку переданного ей массива.

Написать функции для работы со стеком: инициализация стека, запись в стек и чтение из стека. Стек имитируется как массив заданного размера с первоначальным значением указателя вершины стека, равным размеру массива.

Вывести на консоль линейку меню: 1, 2, 3, 4 с приглашением выбора. Через массив указателей на функции организовать реакцию на выбор пользователя (вывод на консоль соответствующей строки).

Задания открытого типа

1. Операторы.
2. Комментарии.
3. Интегрированная среда разработки.
4. Архитектурные типы.
5. Основные компоненты: поиск и навигация, редактирование и рефакторинг, отладка и тестирование.
6. Система управления версиями.
7. Теорема Бёма — Якопини.
8. Структура программы.
9. Комментарии.
10. Стандартный ввод/вывод.
11. Конструкция следования. Стандартные функции языка.
12. Организация ветвлений: бинарное и множественное ветвление.
13. Тернарная операция.
14. Множественный выбор.
15. Организация циклов.
16. Конструкторы.
17. Указатель `this`.
18. Статические элементы класса.
19. Расширение интерфейса класса.
20. Деструкторы.

21. Операции над строками.
22. Доступ к элементу строки.
23. Функции для работы со строками: присваивание и добавление частей строк, получение характеристик строк, преобразование строк, поиск подстрок, сравнение частей строк.
24. Последовательные контейнеры.
25. Общие методы контейнеров.

ОПК-4

Типовые тесты

Вопрос №1. Укажите формы представления алгоритма:

Тип ответа: Многие из многих

Варианты ответов:

1. Запись на естественном языке
2. Язык программирования
3. Язык стилистической разметки документа
4. Блок-схемы
5. Псевдокод

Вопрос №2. С помощью каких управляющих структур можно представить алгоритм любой сложности:

Тип ответа: Многие из многих

Варианты ответов:

1. Вспомогательный алгоритм
2. Ветвление
3. Следование
4. Безусловный переход
5. Цикл

Вопрос №3. В языке блок-схем какой символ (фигура) используется для организации функции переключательного типа, в которой активизируется только один из альтернативных выходов после вычисления условия:

Варианты ответов:

1. Ромб
2. Трапеция
3. Прямоугольник
4. Параллелограмм

Вопрос №4. В языке блок-схем какой символ (фигура) отображает функцию обработки данных любого вида:

Варианты ответов:

1. Ромб
2. Трапеция
3. Прямоугольник
4. Прямоугольник

Вопрос №5. Как определяет понятие «программа» ГОСТ 19781-90:

Варианты ответов:

1. Система обработки информации и документов, необходимых для эксплуатации программы
2. Данные, предназначенные для управления конкретными компонентами системы

обработки информации в целях реализации определенного алгоритма

3. Система инженерных принципов, направленных на создание алгоритма, реализующего обработку данных

4. Система принципов и способов организации процесса обработки данных

Практическое задание

Лабораторная работа

№1

Рассмотрите файл F0.

Файл F0

A	B	C	D	E
A1	b1	c1	d1	e2
A2	b2	c2	d1	e2
A3	b3	c3	d2	e5
A4	b1	c4	d3	e2

Далее выполните следующие действия:

1. Создайте соответствующую таблицу в базе данных СУБД Access

2. Реализуйте средствами SQL запросов СУБД Access следующие проекции:

- проекцию файла F0 на столбцы D и E;
- все проекции файла F0 на один реквизит;
- все проекции файла F0 на три реквизита;

№2

Разработайте структуру двухуровневой сетевой базы данных для заданного множества реквизитов и отношений:

Реквизиты	Отношения
ФИО студента	W(ФИО, Группа)
Группа	S(Преподаватель, Кафедра)
Дисциплина	Z(ФИО, Дисциплина, Преподаватель, Оценка)
Преподаватель	Y(Группа, Преподаватель, дисциплина, День)
Кафедра	F(Дисциплина, Кафедра)
Оценка экзамена	
День занятий	

Студенты и преподаватели-однофамильцы отсутствуют.

Задания открытого типа

1. Вложенные циклы.
2. Принудительное завершение итерации и цикла.
3. Описание и определение функций. Вызов функции. Возвращаемое значение функции.
4. Параметры функции: позиционные, по умолчанию, именованные.
5. Функция с переменным числом параметров.
6. Передача параметров по значению и по адресу.
7. inline-функции.
8. Использование глобальных переменных.
9. Рекурсивная функция.

10. Перегрузка функций. Правила описания перегруженных функций. Схема алгоритма поиска функции.
11. Одномерные и многомерные массивы. Определение и инициализация.
12. Доступ к элементу массива. Пакетная обработка массива.
13. Линейный и бинарный поиск в массиве.
14. Модификация массивов.
15. Передача массивов в функцию.
16. Массивы символов.
17. Объектно-ориентированное программирование.
18. Классы. Описание класса.
19. Члены класса. Управление доступом к членам класса.
20. Экземпляры класса. Доступ к элементам экземпляра.
21. Индексируемая коллекция.
22. Класс Array. Одномерные и многомерные массивы: объявление и инициализация.
23. Операции над строками.
24. Подсистема обработки исключений.
25. Сериализация.

ОПК-7

Типовые тесты

1. Содержимое переменной d после выполнения следующего кода:

```
int a = 3, b=0, c = -7, d=0;
```

```
if(a)
```

```
{
```

```
if(b) d=1;
```

```
if(c) d = 2;
```

```
else d =3;
```

```
}
```

```
else d=4;
```

Варианты ответов:

1. 1
2. 2
3. 3
4. 4

2. Какую задачу выполняет следующий фрагмент кода:

```
int mas[10] = {3, 5, 7, 8, 2, 4, 6, 9, 10, 1};
```

```
int n = mas[0];
```

```
for(int i = 0; i<10; i++) if(m[i] < n) n = m[i];
```

Варианты ответов:

1. сортировка
2. нахождение максимального элемента массива
3. нахождение минимального элемента массива
4. нахождение суммы элементов массива

3. Сумма элементов массива s после выполнения следующего кода:

```
int m[]={1, 2, 3, 4, -5, 6, -7, 8, 9, 10};
```

```
int s =0;
```

```
for(int i =0; i <10 && m[i]>0; i++) s+=m[i];
```

Варианты ответов:

1. 31
2. 55

3. 43

4. 10

4. Заданы структурная переменная a и массив m: `struct A {int x; int y};`

`A a = {0,0};`

`int m[5] = {0};`

Среди ниже перечисленных выражений, осуществляющих присваивание значений элементам объектов, укажите неверное

Варианты ответов:

1. `a[0] = 1;`

2. `m[0] = 1;`

3. `a.x = 1;`

4. `*(m + 0) = 1;`

5. Символ, применяемый в C++ для обозначения конца строки

Варианты ответов:

1. `'\t'`

2. `'\n'`

3. `'\r'`

4. `'\0'`

Практическое задание Лабораторная работа №1

Пусть имеются отношения T1 и T2.

T1	A	B	C		T2	A	D
	a1	8	4			a1	5
	a2	1	3			a2	4
	a3	6	5			a4	2
	a1	2	4				

Выполните следующие соединения:

1. $T3 = T1[B < D] T2;$

2. $T4 = T1[C > D] T2$

№2

Рассмотрите приведенные ниже факты и правила:

Правила:

1. если ЦЕЛЬ - ОТДЫХ и ДОРОГА ПЛОХАЯ то ИСПОЛЬЗОВАТЬ ДЖИП.

2. если МЕСТО_ОТДЫХА - ГОРЫ то ДОРОГА ПЛОХАЯ.

3. если МЕСТО_ОТДЫХА - ПЛЯЖ то ДОРОГА ПЛОХАЯ.

Факты:

1. ЦЕЛЬ - ОТДЫХ.

2. МЕСТО_ОТДЫХА - ГОРЫ.

3. ДОРОГА ПЛОХАЯ.

4. МЕСТО_ОТДЫХА - ПЛЯЖ.

5. ИСПОЛЬЗОВАТЬ ДЖИП.

Далее выполните следующие действия:

- установите последовательность применяемых правил и новых фактов в процессе работы механизма прямого вывода для приведенных фактов и правил, если первоначально в

рабочей памяти хранятся факты с номерами 1 и 2;
- установите последовательность новых целей и их доказательств в процессе работы механизма обратного вывода для приведенных фактов и правил, если необходимо доказать цель с номером 5

Задания открытого типа

1. Сортировка. Классификация сортировок.
2. Основные виды сортировок: пузырьковая, выбором, шейкерная, расческой.
3. Эффективные сортировки: быстрая, слиянием.
4. Указатели. Виды указателей.
5. Способы инициализации указателей.
6. Операции с указателями.
7. Арифметические операции с указателями.
8. Ссылки. Правила инициализации ссылок. Виды ссылок.
9. Ссылки на массив и на функцию.
10. Динамически распределяемая память.
11. Выделение динамической памяти: оператор new.
12. Освобождение динамической памяти: оператор delete.
13. Указатели и динамическая память.
14. Массивы в динамической памяти.
15. Передача динамических массивов в функции.
16. Отслеживание утечек памяти.
17. Перечисляемый тип.
18. Структура. Инициализация структуры.
19. Элементы структуры. Доступ к полям структуры. Битовые поля.
20. Объединения. Ограничения, накладываемые на объединения.
21. Варианты реализации отношения "клиент-поставщик": создание объекта класса поставщика, вызов статических полей или методов класса поставщика.
22. Управление доступом для методов set и get.
23. Множественное наследование. Интерфейсы.
24. Свойства делегата.
25. Подписка на события. Подписка на события с помощью анонимной функции.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Специфика формирования компетенций и их измерение определяется структурированием информации о состоянии уровня подготовки обучающихся.

Алгоритмы отбора и конструирования заданий для оценки достижений в предметной области, техника конструирования заданий, способы организации и проведения стандартизированных оценочных процедур, методика шкалирования и методы обработки и интерпретации результатов оценивания позволяют обучающимся освоить компетентностно-ориентированные программы дисциплин.

Формирование компетенций осуществляется в ходе всех видов занятий, практики, а контроль их сформированности на этапе текущей, промежуточной и итоговой аттестации.

Оценивание знаний, умений и навыков по учебной дисциплине осуществляется посредством использования следующих видов оценочных средств:

- опросы: устный, письменный;
- задания для практических занятий;
- ситуационные задания;

- контрольные работы;
- коллоквиумы;
- написание реферата;
- написание эссе;
- решение тестовых заданий;
- экзамен.

Опросы по вынесенным на обсуждение темам

Устные опросы проводятся во время практических занятий и возможны при проведении аттестации в качестве дополнительного испытания при недостаточности результатов тестирования и решения заданий. Вопросы опроса не должны выходить за рамки объявленной для данного занятия темы. Устные опросы необходимо строить так, чтобы вовлечь в тему обсуждения максимальное количество обучающихся в группе, проводить параллели с уже пройденным учебным материалом данной дисциплины и смежными курсами, находить удачные примеры из современной действительности, что увеличивает эффективность усвоения материала на ассоциациях.

Основные вопросы для устного опроса доводятся до сведения студентов на предыдущем практическом занятии.

Письменные опросы позволяют проверить уровень подготовки к практическому занятию всех обучающихся в группе, при этом оставляя достаточно учебного времени для иных форм педагогической деятельности в рамках данного занятия. Письменный опрос проводится без предупреждения, что стимулирует обучающихся к систематической подготовке к занятиям. Вопросы для опроса готовятся заранее, формулируются узко, дабы обучающийся имел объективную возможность полноценно его осветить за отведенное время.

Письменные опросы целесообразно применять в целях проверки усвояемости значительного объема учебного материала, например, во время проведения аттестации, когда необходимо проверить знания обучающихся по всему курсу.

При оценке опросов анализу подлежит точность формулировок, связность изложения материала, обоснованность суждений.

Решение заданий (кейс-методы)

Решение кейс-методов осуществляется с целью проверки уровня навыков (владений) обучающегося по применению содержания основных понятий и терминов дисциплины вообще и каждой её темы в частности.

Обучающемуся объявляется условие задания, решение которого он излагает либо устно, либо письменно.

Эффективным интерактивным способом решения задания является сопоставления результатов разрешения одного задания двумя и более малыми группами обучающихся.

Задачи, требующие изучения значительного объема, необходимо относить на самостоятельную работу студентов, с непременно разбором результатов во время практических занятий. В данном случае решение ситуационных задач с глубоким обоснованием должно представляться на проверку в письменном виде.

При оценке решения заданий анализируется понимание обучающимся конкретной ситуации, правильность её понимания в соответствии с изучаемым материалом, способность обоснования выбранной точки зрения, глубина проработки рассматриваемого вопроса, умением выявить основные положения затронутого вопроса.

Решение заданий в тестовой форме

Проводится тестирование в течение изучения дисциплины

Не менее чем за 1 неделю до тестирования, преподаватель должен определить обучающимся исходные данные для подготовки к тестированию: назвать разделы (темы,

вопросы), по которым будут задания в тестовой форме, теоретические источники (с точным указанием разделов, тем, статей) для подготовки.

При прохождении тестирования пользоваться конспектами лекций, учебниками, и иными материалами не разрешено.