

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Белгородский Валерий Савельевич
Должность: Ректор
Дата подписания: 25.06.2024 15:24:16
Уникальный программный ключ:
8df276ee93e17c18e7bee9e7cad2d0ed9abb2479

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный университет им. А.Н. Косыгина
(Технологии. Дизайн. Искусство)»

Институт экономики и менеджмента

Кафедра информационных технологий

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Технология программирования

Уровень образования	бакалавриат
Направление подготовки	38.03.05 Бизнес-информатика
Профиль	Бизнес-информатика в экономике
Срок освоения образовательной программы по очной форме обучения	4 года
Форма обучения	очная

Рабочая программа учебной дисциплины «Технология программирования» основной профессиональной образовательной программы высшего образования рассмотрена и одобрена на заседании кафедры, протокол № 9 от 11.04.2024 г.

Разработчики рабочей программы «Технология программирования»:

1. Канд. техн. наук, доцент А. А. Семенов
2. Старший преподаватель А. М. Козлов

Заведующий кафедрой: канд. техн. наук, доцент И. Б. Разин

1. ОБЩИЕ СВЕДЕНИЯ

Учебная дисциплина «Технология программирования» изучается в третьем, четвертом семестрах.

Курсовая работа/Курсовой проект – не предусмотрены.

1.1. Формы промежуточной аттестации:

- третий семестр - экзамен
- четвертый семестр - экзамен

1.2. Место учебной дисциплины в структуре ОПОП

Учебная дисциплина «Технология программирования» относится к обязательной части.

Основой для освоения дисциплины являются результаты обучения по предшествующим дисциплинам и практикам:

– Информационные и коммуникационные технологии в профессиональной деятельности.

Результаты обучения по учебной дисциплине, используются при изучении следующих дисциплин:

- Технологии обработки информации;
- Методы и средства проектирования информационных систем и технологий.

2. ЦЕЛИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Целями изучения дисциплины «Технология программирования» являются:

- изучение способов представления и структурирования информации о явлениях и процессах в окружающем мире применительно к своей профессиональной деятельности;
- освоение методов ориентирования и взаимодействия с ресурсами информационной среды, осуществления выбора различных моделей использования информационных и коммуникационных технологий в экономике и бизнесе;
- изучение методов построения алгоритмов и основных этапов разработки и создания современных программных продуктов;
- освоение подходов к построению рациональных диалоговых интерфейсов, ориентированных на пользователя;
- изучение базовых правил и принципов современного объектно-ориентированного и визуального программирования;
- формирование у обучающихся компетенций, установленных образовательной программой в соответствии с ФГОС ВО по данной дисциплине.

Результатом обучения по учебной дисциплине является овладение обучающимися знаниями, умениями, навыками и опытом деятельности, характеризующими процесс формирования компетенций и обеспечивающими достижение планируемых результатов освоения учебной дисциплины.

2.1. Формируемые компетенции, индикаторы достижения компетенций, соотнесённые с планируемыми результатами обучения по дисциплине:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине
--------------------------------	--	---

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине
ОПК-3 Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации	ИД-ОПК-3.1 Применение приемов проектирования и разработки различных видов программного обеспечения с использованием современных ИТ	<ul style="list-style-type: none"> - Оценивает сущность и значение информационных технологий в развитии современного общества; владеет основными методами, способами и средствами получения, хранения, переработки информации; - Использует навыки работы с программными средствами для управления информацией и коммуникации с соблюдением основных требований информационной безопасности. - Применяет методики использования программных средств для решения практических задач в профессиональной деятельности; - Умеет разрабатывать современные эффективные интерфейсы «человек - электронно-вычислительная машина»; - Рационально оценивает и обосновывает принимаемые проектные решения, составляет блок-схемы алгоритмов и реализует их на языках программирования высокого уровня; - Осуществляет практическую реализацию информационно-коммуникационных программ и компьютерных приложений, выполняет тестовые примеры для проверки их корректности и эффективности.
	ИД-ОПК-3.2 Применение приемов разработки регламентов бизнес-процессов в целях управления процессами создания и использования продуктов и услуг в сфере ИКТ	

3. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Общая трудоёмкость учебной дисциплины по учебному плану составляет:

по очной форме обучения –	9	з.е.	288	час.
---------------------------	---	------	-----	------

3.1. Структура учебной дисциплины для обучающихся по видам занятий (очная форма обучения)

Структура и объем дисциплины									
Объем дисциплины по семестрам	форма промежуточной аттестации	всего, час	Контактная аудиторная работа, час				Самостоятельная работа обучающегося, час		
			лекции, час	практические занятия, час	лабораторные занятия, час	практическая подготовка, час	курсовая работа/ курсовой проект	самостоятельная работа обучающегося, час	промежуточная аттестация, час
3 семестр	экзамен	160	18		54	9		47	32
4 семестр	экзамен	128	18		36	10		32	32

Bcero:		288	36		90	19		79	64
--------	--	-----	----	--	----	----	--	----	----

3.2. Структура учебной дисциплины для обучающихся по разделам и темам дисциплины: (очная форма обучения)

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы/ индивидуальные занятия, час	Практическая подготовка, час		
Третий семестр							
ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2	Раздел I. Основные понятия программирования.	3	х	12	2	11	
	Лекция 1.1. Вычислительные системы и периферийные устройства.	1				х	Контроль посещаемости.
	Лекция 1.2. Алгоритмы и их свойства.	2				х	Контроль посещаемости.
	Лабораторная работа № 1.1. Иерархическая система классификации информации.			6	1	5	Выполнение лабораторной работы.
	Лабораторная работа № 1.2. Составление блок-схем алгоритмов.			6	1	6	Выполнение лабораторной работы.
ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2	Раздел II. Языки программирования высокого уровня.	3	х	12	2	11	
	Лекция 2.1. Структура программы на языках высокого уровня.	1				х	Контроль посещаемости.
	Лекция 2.2. Основные операторы языков высокого уровня	2				х	Контроль посещаемости.
	Лабораторная работа № 2.1. Вычисления и расчёты с целыми типами данных.			6	1	5	Выполнение лабораторной работы.
	Лабораторная работа № 2.2. Вычисления и расчёты с вещественными типами данных.			6	1	6	Выполнение лабораторной работы.
ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2	Раздел III. Структурные типы данных.	6	х	18	3	11	
	Лекция 3.1. Символьные и строковые типы данных.	2				х	Контроль посещаемости.
	Лекция 3.2. Тип-диапазон, перечисляемые типы, массивы.	2				х	Контроль посещаемости.
	Лекция 3.3. Множества, записи.	2				х	Контроль посещаемости.
	Лабораторная работа № 3.1. Работа с символьными данными.			6	1	4	Выполнение лабораторной работы.
	Лабораторная работа № 3.2. Работа с матрицами (двумерными массивами).			6	1	4	Выполнение лабораторной работы.

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы/индивидуальные занятия, час	Практическая подготовка, час		
	Лабораторная работа № 3.3. Работа с записями.			6	1	3	Выполнение лабораторной работы.
ОПК-3:	Раздел IV. Подпрограммы.	6	x	12	2	14	
ИД-ОПК-3.1	Лекция 4.1. Описание подпрограмм. Параметры подпрограмм.	2				x	Контроль посещаемости.
ИД-ОПК-3.2	Лекция 4.2. Структура модулей. Интерфейсная и исполняемая часть модулей.	2				x	Контроль посещаемости.
	Лекция 4.3. Процедурный тип данных. Создание и использование библиотек DLL.	2				x	Контроль посещаемости.
	Лабораторная работа № 4.1. Использование подпрограмм и модулей.			6	1	7	Выполнение лабораторной работы.
	Лабораторная работа № 4.2. Использование библиотек DLL.			6	1	7	Выполнение лабораторной работы.
	Экзамен	x	x	x	x	32	Электронное тестирование.
	ИТОГО за третий семестр	18		54	9	79	Экзамен
	Четвертый семестр						
ОПК-3:	Раздел V. Введение в C++	1	x	2	x	3	
ИД-ОПК-3.1	Лекция 5.1. Характеристика языка программирования C++ и структура программы на C++	1					Контроль посещаемости.
ИД-ОПК-3.2	Лабораторная работа № 5.1. Линейное программирование			2		3	Выполнение лабораторной работы.
ОПК-3:	Раздел VI. Переменные и функции стандартной библиотеки	2	x	2	1	3	
ИД-ОПК-3.1	Лекция 6.1. Переменные и функции стандартной библиотеки C++	2					Контроль посещаемости.
ИД-ОПК-3.2	Лабораторная работа № 6.1. Расчетные задачи			2	1	3	Выполнение лабораторной работы.
ОПК-3:	Раздел VII. Базовые конструкции программирования и	2	x	2	1	3	

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы/индивидуальные занятия, час	Практическая подготовка, час		
ИД-ОПК-3.1 ИД-ОПК-3.2	циклы						
	Лекция 7.1. Базовые конструкции программирования и циклы	2					Контроль посещаемости.
	Лабораторная работа № 7.1. Циклы			2	1	3	Выполнение лабораторной работы.
ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2	Раздел VIII. Ветвления и передача управления	1	x	4	1	3	
	Лекция 8.1. Ветвления и передача управления	1					Контроль посещаемости.
	Лабораторная работа № 8.1. Ветвления и циклы			4	1	3	Выполнение лабораторной работы.
ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2	Раздел IX. Массивы и строки	2	x	4	2	4	
	Лекция 9.1. Массивы и строки	2					Контроль посещаемости.
	Лабораторная работа № 9.1. Массивы			2	1	2	Выполнение лабораторной работы.
	Лабораторная работа № 9.2. Строки типа char			2	1	2	
ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2	Раздел X. Функции	2	x	4	1	4	
	Лекция 10.1. Функции	2					Контроль посещаемости.
	Лабораторная работа № 10.1. Функции			4	1	4	Выполнение лабораторной работы.
ОПК-3:	Раздел XI. Указатели и динамическое выделение памяти	2	x	4	1	3	

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы/индивидуальные занятия, час	Практическая подготовка, час		
ИД-ОПК-3.1 ИД-ОПК-3.2	Лекция 11.1. Указатели и динамическое выделение памяти	2					Контроль посещаемости.
	Лабораторная работа № 11.1. Указатели и функции. Динамическое выделение памяти			4	1	3	Выполнение лабораторной работы.
ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2	Раздел XII. Потoki и файлы	2	x	4	1	3	
	Лекция 12.1. Потoki и файлы	2					Контроль посещаемости.
	Лабораторная работа № 12.1. Потокoвые классы C++ и файлы. Текстовые и бинарные файлы			4	1	3	Выполнение лабораторной работы.
ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2	Раздел XIII. Технологии программирования. Структуры	4	x	10	2	6	
	Лекция 13.1. Технологии программирования. Структуры	4					Контроль посещаемости.
	Лабораторная работа № 13.1. Массивы и функции			4		2	Выполнение лабораторной работы.
	Лабораторная работа № 13.2. Простая графика			2	1	2	Выполнение лабораторной работы.
	Лабораторная работа № 13.3. Графика. Структура.			4	1	2	Выполнение лабораторной работы.
	Экзамен	x	x	x	x	32	Экзамен по билетам
	ИТОГО за четвертый семестр	18		36	10	64	Экзамен
	ИТОГО за весь период	36		90	19	143	

3.3. Краткое содержание учебной дисциплины

№ пп	Наименование раздела и темы дисциплины	Содержание раздела (темы)
Третий семестр		
Раздел I Основные понятия программирования		
Лекция 1.1	Вычислительные системы и периферийные устройства.	Введение. Основные понятия устройства вычислительных систем: процессор, оперативная память, запоминающие устройства. Классификация периферийных устройств для ввода и вывода информации. Принципы функционирования периферийных устройств.
Лекция 1.2	Алгоритмы и их свойства.	Важность формирования алгоритмического мышления для решения инженерных задач. Различные подходы к понятию «алгоритм». Понятие исполнителя алгоритма. Основные свойства алгоритмов. Представление алгоритмов: математическое, словесное, графическое, записанное на алгоритмическом языке, на языке программирования и пр. Основные элементы блок-схем для изображения алгоритмов. Принципы разработки алгоритмов: операциональный, структурный, объектно-ориентированный. Современные методологии разработки программ для ЭВМ.
Лабораторная работа № 1.1	Иерархическая система классификации информации.	Понятия форм и окон в графических операционных системах. Визуальные компоненты в современных средах разработки приложений с графическими интерфейсами. Свойства и методы объектов и классов. Объявление объектов в разделе описаний и создание методов в виде подпрограмм. Инициализация полей объектов. Использование объектов и классов в современных средах разработки приложений. Библиотеки объектов, классов.
Лабораторная работа № 1.2	Составление блок-схем алгоритмов.	Обсуждение лекции и способов выполнения лабораторной работы. Составление блок-схем алгоритмов для математических задач и для различных процессов человеческой деятельности. Разработка блок-схемы алгоритма решения математической задачи согласно вариантам заданий. Составление описания последовательности действий в виде блок-схемы процесса для абстрактного Исполнителя.
Раздел II Языки программирования высокого уровня.		
Лекция 2.1	Структура программы на языках высокого уровня	Введение. Основные понятия программирования. Структура программы на языках-компиляторах в общем виде с модулями, функциями и процедурами. Разделы объявлений типов и разделы исполняемых операторов для главной программы, модулей и подпрограмм. Изучение программной оболочки Lazarus, компилятора и основных операторов языка Object Pascal. Создание консольных приложений. Создание современных прикладных программ методами визуального программирования. Конструктор формы и визуальные компоненты. Свойства и методы объектов. Написание своего программного кода. Отладка и компиляция программ.
Лекция 2.2	Основные операторы языков высокого уровня	Основные операторы, присваивание значений переменным, арифметические операции, условный оператор, циклы с параметром и с условием. Использование составных логических условий в

		операторах ветвления и циклов. Типы данных и их представление в двоичном коде. Различие операций с разными типами данных. Основные операторы и операции для работы с разными типами. Важность использования функций преобразования типов и форматирования в операциях ввода-вывода. Целесообразность и эффективность при выборе типов данных. Порядковые типы данных. Логические типы данных. Целые типы данных. Вещественные типы данных.
Лабораторная работа № 2.1	Вычисления и расчёты с целыми типами данных.	Использование арифметических операций, процедур и функций для работы с целыми и вещественными типами данных согласно вариантам заданий.
Лабораторная работа № 2.2	Вычисления и расчёты с вещественными типами данных.	Ввод данных для тригонометрических вычислений согласно вариантам заданий. Расчёты по соответствующим формулам и вывод результатов.
Раздел III	Структурные типы данных	
Лекция 3.1	Символьные и строковые типы данных.	Однобайтные и двухбайтные символьные данные. Операции с символьными данными. Связь символов и их кодов в различных кодовых таблицах. Строковые типы данных как массивы символов. Операторы для работы со строками. Преобразования строк разных типов. Изучение методик алгоритмизации при работе с многомерными массивами: индексация элементов, поиск элементов в строках и столбцах.
Лекция 3.2	Тип-диапазон, перечисляемые типы, массивы.	Использование типа-диапазона в операторах множественного выбора и при объявлении массивов. Применение перечисляемых типов данных для работы с ассоциативными массивами. Массивы как способ создания последовательных списков однородных данных. Диапазоны индексов массивов. Обращение к элементам массивов. Особенности использования одномерных и многомерных массивов.
Лекция 3.3	Множества, записи.	Особенности операций с типом-множество. Количество возможных присваиваемых значений переменным типа множество. Примеры программ, использующих тип-множество. Записи как основной способ объединения и структурирования данных разных типов. Объявление записей и их использование в программах. Доступ к полям записей. Вариантные поля записей. Изучение методик алгоритмизации при работе с записями: разработка рациональных структурированных данных с разнотипными полями, создание списков записей, обработка записей для составления выборок информации.
Лабораторная работа № 3.1	Работа с символьными данными.	Обработка символьных строк в соответствии с вариантами заданий. Анализ и преобразования символьных данных.
Лабораторная работа № 3.2	Работа с матрицами (двумерными массивами).	Ввод данных в двумерный массив (матрицу). Обработка данных (математические вычисления в строках и столбцах матрицы) в соответствии с вариантами заданий.
Лабораторная работа	Работа с записями.	Ввод структурированной информации в поля записей. Создание выборок информации по заданным полям в

работа № 3.3		соответствии с вариантами заданий и вывод результатов вычислений.
Раздел IV	Подпрограммы	
Лекция 4.1	Описание подпрограмм. Параметры подпрограмм.	Описание подпрограмм в соответствующих разделах языка высокого уровня. Параметры подпрограмм: параметры-значения, параметры-переменные, параметры-константы. Выбор способа передачи параметров в подпрограмму и возвращения результатов. Различные способы вызова подпрограмм. Рекурсия и опережающее описание.
Лекция 4.2	Структура модулей. Интерфейсная и исполняемая часть модулей.	Модульная структура приложений. Понятие «проект» в современных средах разработки программ на языках высокого уровня. Создание файла-модуля. Структура модуля. Интерфейсная и исполняемая часть модулей. Доступ к объявленным в модуле подпрограммам. Подключение модулей к головной программе и вызов подпрограмм. Изучение возможностей Object Pascal для написания и использования модулей.
Лекция 4.3	Процедурный тип данных. Создание и использование библиотек DLL	Создание и использование процедурных типов данных. Тип-процедура и тип-функция. Передача подпрограммы в качестве параметра в другую подпрограмму. Вызов внешних подпрограмм из модулей и библиотек. Адрес подпрограммы и его получение для доступа и вызова подпрограммы. Способы создания файлов на языках высокого уровня для их компиляции в динамически подключаемые библиотеки. Особенности использования библиотек в различных операционных системах. Статический и динамический способы вызова подпрограмм из библиотек. Основные операторы для работы с DLL в операционных системах семейства Windows. Изучение возможностей среды разработки Lazarus для написания и использования библиотек DLL.
Лабораторная работа № 4.1	Использование подпрограмм и модулей	Оформление подпрограммы в отдельном модуле. Подключение модуля к головной программе и использование подпрограммы. Тип подпрограммы и способ обработки данных указаны в вариантах заданий.
Лабораторная работа № 4.2	Использование библиотек DLL.	Оформление подпрограммы с заданиями по вариантам в библиотеках DLL. Вызов подпрограмм из головной программы статическим и динамическим способами.
Четвертый семестр		
Раздел VII	Введение в C++	
Лекция 7.1	Характеристика языка программирования C++ и структура программы на C++.	Характеристика языка C++. Состав языка и принцип работы компилятора языка высокого уровня. Идентификаторы. Типы данных в C++. Структура программы на языке C++. Создание консольных и визуальных приложений в ИСР C++Builder. Альтернативный ввод-вывод в стиле C.
Лабораторная работа № 7.1	Линейное программирование.	Обзор сред разработки на C++. Изучение основных инструментов интегрированной среды разработки C++Builder, разработка консольных и визуальных проектов. Реализация задач линейного программирования согласно вариантам заданий.

Раздел VIII	Переменные и функции стандартной библиотеки	
Лекция 8.1	Переменные и функции стандартной библиотеки C++	Переменные в C++. Константы. Манипуляторы. Преобразования типов. Арифметические операции. Функции стандартной библиотеки.
Лабораторная работа № 8.1	Расчетные задачи	Применение арифметических операций для работы с целыми и вещественными типами данных. Реализация расчетных задач согласно вариантам заданий.
Раздел IX	Базовые конструкции программирования и циклы	
Лекция 9.1	Базовые конструкции программирования и циклы	Операции отношения и логические операции. Базовые конструкции структурного программирования. Циклы. Цикл с параметром (for). Цикл с предусловием (while). Цикл с постусловием (do while). Рекомендации по программированию циклов.
Лабораторная работа № 9.1	Циклы	Применение циклов для реализации расчетных задач согласно вариантам заданий.
Раздел X	Ветвления и передача управления	
Лекция 10.1	Ветвления и передача управления	Ветвления. Условный оператор if. Вложенные конструкции. Проблемы соответствия if и else во вложенных ветвлениях. Условная операция. Оператор switch. Операторы передачи управления. Приоритеты операций в C++.
Лабораторная работа № 10.1	Ветвления и циклы	Циклические конструкции с условиями и оператором switch. Применение ветвлений и циклов для реализации расчетных задач согласно вариантам заданий.
Раздел XI	Массивы и строки	
Лекция 11.1	Массивы и строки	Массивы. Сортировки массивов. Многомерные массивы. Векторы. Строки типа char. Операции со строками.
Лабораторная работа № 11.1	Массивы	Реализация операций с одномерными и двумерными массивами согласно вариантам заданий.
Лабораторная работа № 11.2	Строки типа char	Реализация операций со строками типа char согласно вариантам заданий.
Раздел XII	Функции	
Лекция 12.1	Функции	Виды функций в C++. Передача аргументов в функцию по значению. Возвращение функцией значения. Передача аргументов по ссылке. Возвращение значения по ссылке. Перегруженные функции. Переменные и функции. Передача массивов в функции.
Лабораторная работа № 12.1	Функции	Процедурное программирование. Применение функций при разработке консольных и визуальных приложений согласно вариантам заданий.
Раздел XIII	Указатели и динамическое выделение памяти	

Лекция 13.1	Указатели и динамическое выделение памяти	Характеристика и синтаксис указателей в C++. Указатели и массивы. Указатели и функции. Указатели и строки. Динамический массив и динамическое выделение памяти.
Лабораторная работа № 13.1	Указатели и функции. Динамическое выделение памяти	Применение указателей при передаче аргументов в функции на примере обработки одномерных и двумерных динамических массивов при разработке консольных и визуальных приложений согласно вариантам заданий.
Раздел XIV	Потоки и файлы	
Лекция 14.1	Потоки и файлы	Ввод/вывод. Потоки C++ и файлы. Форматированный ввод/вывод. Конструкторы и методы. Двоичный ввод/вывод. Бинарные файлы. Двухнаправленный ввод/вывод. Функции стандартной библиотеки для работы с файлами или ввод/вывод в стиле C.
Лабораторная работа № 14.1	Потоковые классы C++ и файлы. Текстовые и бинарные файлы	Работа с текстовыми и бинарными файлами при разработке консольных и визуальных приложений согласно вариантам заданий.
Раздел XV	Технологии программирования. Структуры	
Лекция 15.1	Технологии программирования. Структуры	Технологии программирования. Правила кодирования и документирования программы. Структуры. Структуры и функции. Массивы структур. Поиск в массиве структур. Вложенность структур. Рекурсия. Алгоритм быстрой сортировки. Массивы структур и бинарные файлы.
Лабораторная работа № 15.1	Массивы и функции	Обработка массивов посредством функций при разработке консольных и визуальных приложений согласно вариантам заданий.
Лабораторная работа № 15.2	Простая графика	Работа с компонентом Image. Построение графических примитивов: отрезок, круг, сектор, эллипс, прямоугольник, полигон, полилиния. Рисование на холсте с динамической закраской. Выбор цвета, толщины и формы кисти.
Лабораторная работа № 15.3	Графика. Структура. Периметр и площадь	Работа с графикой с применением структуры фигур: <pre>struct Koordinata { int x; int y; };</pre> Построение многоугольников по точкам, сохранение их в текстовый файл, очистка холста, загрузка данных из файла и восстановление нарисованных многоугольников. Расчет периметра и площади нарисованных и загруженных из файла многоугольников.

3.4. Организация самостоятельной работы обучающихся

Самостоятельная работа студента – обязательная часть образовательного процесса, направленная на развитие готовности к профессиональному и личностному самообразованию, на проектирование дальнейшего образовательного маршрута и профессиональной карьеры.

Самостоятельная работа обучающихся по дисциплине организована как совокупность аудиторных и внеаудиторных занятий и работ, обеспечивающих успешное освоение дисциплины.

Аудиторная самостоятельная работа обучающихся по дисциплине выполняется на учебных занятиях под руководством преподавателя и по его заданию. Аудиторная

самостоятельная работа обучающихся входит в общий объем времени, отведенного учебным планом на аудиторную работу, и регламентируется расписанием учебных занятий.

Внеаудиторная самостоятельная работа обучающихся – планируемая учебная, научно-исследовательская, практическая работа обучающихся, выполняемая во внеаудиторное время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия, расписанием учебных занятий не регламентируется.

Внеаудиторная самостоятельная работа обучающихся включает в себя:

- подготовку к лекциям, практическим занятиям, лабораторным работам и экзамену;
- изучение специальной рекомендованной литературы;
- изучение разделов/тем, не выносимых на лекции и практические занятия самостоятельно;
- подготовка к выполнению лабораторных работ;
- подготовка к практическим занятиям;
- подготовка к компьютерному тестированию на промежуточных аттестациях;
- выполнение индивидуальных заданий;
- подготовка к промежуточной аттестации в течение семестра.

Самостоятельная работа обучающихся с участием преподавателя в форме иной контактной работы предусматривает групповую и (или) индивидуальную работу с обучающимися и включает в себя:

- проведение индивидуальных и групповых консультаций по отдельным темам/разделам дисциплины;
- проведение консультаций перед экзаменом, перед зачетом с оценкой;
- консультации по организации самостоятельного изучения отдельных разделов/тем, базовых понятий учебных дисциплин профильного/родственного бакалавриата, которые формировали ОПК и ПК, в целях обеспечения преемственности образования.

Перечень разделов/тем, полностью или частично отнесенных на самостоятельное изучение с последующим контролем:

№ пп	Наименование раздела /темы дисциплины, выносимые на самостоятельное изучение	Задания для самостоятельной работы	Виды и формы контрольных мероприятий (учитываются при проведении текущего контроля)	Трудоемкость, час
Третий семестр				
Раздел I Основные понятия программирования				
Лабораторная работа № 1.1	Иерархическая система классификации информации.	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания	Выполнение лабораторной работы.	5

		лабораторной работы.		
Лабораторная работа № 1.2	Составление блок-схем алгоритмов.	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	6
Раздел II	Языки программирования высокого уровня			
Лабораторная работа № 2.1	Вычисления и расчёты с целыми типами данных.	Изучение учебной, научной и технической литературы по теме лабораторной работы. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения.	Выполнение лабораторной работы.	5
Лабораторная работа № 2.2	Вычисления и расчёты с вещественными типами данных.	Изучение учебной, научной и технической литературы по теме лабораторной работы. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения.	Выполнение лабораторной работы.	6
Раздел III	Структурные типы данных			
Лабораторная работа № 3.1	Работа с символьными данными.	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	4
Лабораторная работа № 3.2	Работа с матрицами (двумерными массивами).	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования.	Выполнение лабораторной работы.	4

		Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.		
Лабораторная работа № 3.3	Работа с записями.	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	3
Раздел IV	Подпрограммы			
Лабораторная работа № 4.1	Использование подпрограмм и модулей	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	7
Лабораторная работа № 4.2	Использование библиотек DLL.	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	7
Третий семестр				
Раздел V	Введение в C++			
Лабораторная работа № 5.1	Линейное программирование.	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной	Выполнение лабораторной работы.	3

		работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.		
Раздел VI	Переменные и функции стандартной библиотеки			
Лабораторная работа № 6.1	Расчетные задачи.	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	3
Раздел VII	Базовые конструкции программирования и циклы			
Лабораторная работа № 7.1	Циклы	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	3
Раздел VIII	Ветвления и передача управления			
Лабораторная работа № 8.1	Ветвления и циклы	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	3
Раздел IX	Массивы и строки			

Лабораторная работа № 9.1	Массивы	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	2
Лабораторная работа № 9.2	Строки типа char	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	2
Раздел X	Функции			
Лабораторная работа № 10.1	Функции	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	4
Раздел XI	Указатели и динамическое выделение памяти			
Лабораторная работа № 11.1	Указатели и функции. Динамическое выделение памяти	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов	Выполнение лабораторной работы.	3

		языка для выполнения задания лабораторной работы.		
Раздел XII	Потоки и файлы			
Лабораторная работа № 12.1	Потоковые классы C++ и файлы. Текстовые и бинарные файлы	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	3
Раздел XIII	Технологии программирования. Структуры			
Лабораторная работа № 13.1	Массивы и функции	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	2
Лабораторная работа № 13.2	Простая графика	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.	Выполнение лабораторной работы.	2
Лабораторная работа № 13.3	Графика. Структура. Периметр и площадь	Изучение научной и технической литературы, нормативных документов, стандартов языков программирования. Работа с материалами конспекта лекций. Анализ задания к лабораторной работе, выбор способов её выполнения. Осваивание методов объектно-ориентированного и	Выполнение лабораторной работы.	2

		визуального программирования. Изучение элементов системы разработки программ и операторов языка для выполнения задания лабораторной работы.		
--	--	---	--	--

3.5. Применение электронного обучения, дистанционных образовательных технологий

При реализации программы учебной дисциплины возможно применение электронного обучения и дистанционных образовательных технологий.

Реализация программы учебной дисциплины с применением электронного обучения и дистанционных образовательных технологий регламентируется действующими локальными актами университета.

Применяются следующие разновидности реализации программы с использованием ЭО и ДОТ.

В электронную образовательную среду, по необходимости, могут быть перенесены отдельные виды учебной деятельности:

использование ЭО и ДОТ	использование ЭО и ДОТ	объем, час	включение в учебный процесс
смешанное обучение	лекции	36	в соответствии с расписанием учебных занятий
	лабораторные занятия	90	

4. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, КРИТЕРИИ ОЦЕНКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ, СИСТЕМА И ШКАЛА ОЦЕНИВАНИЯ

4.1. Соотнесение планируемых результатов обучения с уровнями сформированности компетенций.

Уровни сформированности компетенции(-й)	Итоговое количество баллов в 100-балльной системе по результатам текущей и промежуточной аттестации	Оценка в пятибалльной системе по результатам текущей и промежуточной аттестации	Показатели уровня сформированности		
			универсальной(-ых) компетенции(-й)	общепрофессиональной(-ых) компетенций	профессиональной(-ых) компетенции(-й)
				ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2	
высокий		отлично/ зачтено (отлично)/ зачтено		<p>Обучающийся:</p> <ul style="list-style-type: none"> – исчерпывающе и логически стройно излагает учебный материал, умеет связывать теорию с практикой, справляется с решением задач профессиональной направленности высокого уровня сложности, правильно обосновывает принятые решения; – способен уверенно использовать современные системы разработки прикладных программ с эффективными графическими интерфейсами и системы коммуникации в сети Internet; – показывает творческие способности в понимании и практическом использовании языков высокого уровня, 	

				<p>использовании визуальных компонентов разработки приложений графических интерфейсов;</p> <ul style="list-style-type: none"> – дополняет теоретическую информацию сведениями, самостоятельно полученными из источников научно-технической информации; – способен провести целостный анализ среды разработки современных программ на основе объектно-ориентированного и визуального программирования; – свободно ориентируется в учебной и профессиональной литературе; – дает развернутые, исчерпывающие, профессионально грамотные ответы на вопросы, в том числе, дополнительные. 	
повышенный		хорошо/ зачтено (хорошо)/ зачтено		<p>Обучающийся:</p> <ul style="list-style-type: none"> – достаточно подробно, грамотно и по существу излагает изученный материал, приводит и раскрывает в тезисной форме основные понятия; – анализирует современные Технологии программирования с незначительными пробелами; – способен использовать только основные функциональные возможности систем разработки программ и систем 	

				<p>коммуникации в сети Internet;</p> <ul style="list-style-type: none"> – способен провести анализ основных элементов разработки современных программ на основе объектно-ориентированного и визуального программирования; – допускает единичные негрубые ошибки; – достаточно хорошо ориентируется в учебной и профессиональной литературе; – ответ отражает знание теоретического и практического материала, не допуская существенных неточностей. 	
базовый		удовлетворительно/ зачтено (удовлетворительно)/ зачтено		<p>Обучающийся:</p> <ul style="list-style-type: none"> – демонстрирует теоретические знания основного учебного материала дисциплины в объеме, необходимом для дальнейшего освоения ОПОП; – с неточностями излагает принципы и методы разработки современных программ на основе объектно-ориентированного и визуального программирования; – способен использовать отдельные элементы визуальной разработки прикладных программ; – анализирует современные технологии программирования с неточностями и ошибками; – демонстрирует 	

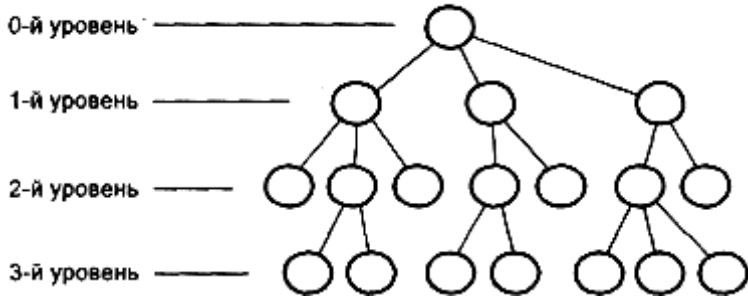
				фрагментарные знания основной учебной литературы по дисциплине; – ответ отражает знания на базовом уровне теоретического и практического материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профилю обучения.
низкий		неудовлетворительно/ не зачтено	Обучающийся: – демонстрирует фрагментарные знания теоретического и практического материал, допускает грубые ошибки при его изложении на занятиях и в ходе промежуточной аттестации; – испытывает серьезные затруднения в применении теоретических положений при решении практических задач профессиональной направленности стандартного уровня сложности, не владеет необходимыми для этого навыками и приёмами; – не способен проанализировать учебно-методическую, техническую и научную литературу; – не владеет основными принципами и навыками работы в современных средах разработки прикладных программ, не умеет пользоваться системами коммуникации (Internet); – выполняет задания только по образцу и под руководством преподавателя; – ответ отражает отсутствие знаний на базовом уровне теоретического и практического материала в объеме, необходимом для дальнейшей учебы.	

5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ, ВКЛЮЧАЯ САМОСТОЯТЕЛЬНУЮ РАБОТУ ОБУЧАЮЩИХСЯ

При проведении контроля самостоятельной работы обучающихся, текущего контроля и промежуточной аттестации по учебной дисциплине «Технология программирования» проверяется уровень сформированности у обучающихся компетенций и запланированных результатов обучения по дисциплине, указанных в разделе 2 настоящей программы.

5.1. Формы текущего контроля успеваемости, примеры типовых заданий:

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
Лабораторная работа № 1.1	Выполнение лабораторной работы.	Задание 1. Иерархическая система классификации.	ОПК-3: ИД-ОПК-3.1

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		 <p>0-й уровень —</p> <p>1-й уровень —</p> <p>2-й уровень —</p> <p>3-й уровень —</p> <p>В текстовом редакторе Microsoft Word создать иерархическую систему классификации объектов согласно вариантам задания. Она должна содержать не менее 4-х уровней (0-3) классификации и не менее 12 конечных объектов. Варианты: 1 – Наземный транспорт. 2 – Воздушный транспорт. 3 – Водный транспорт. 4 – Одежда. 5 – Обувь. ... Задание 2. В текстовом редакторе Microsoft Word описать различия между тремя словами-дескрипторами согласно вариантам. Привести примеры фраз, в которых одно слово не может быть заменено другим. Варианты: 1 – Данные – Сведения - Факты. 2 – Идти – Шагать - Ступать. 3 – Идентичность – Тождественность - Совпадение. 4 – Последовательность – Порядок - Программа. 5 – Выполнение – Реализация - Воплощение. ...</p>	ИД-ОПК-3.2
Лабораторная работа № 1.2	Выполнение лабораторной работы.	Составление блок-схем алгоритмов. Задание 1. Разработка блок-схемы алгоритма решения задачи. Варианты заданий:	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>1. Дан номер месяца (1 – январь, 2 – февраль, ...). Вывести название соответствующего времени года («зима», «весна» и т. д.).</p> <p>2. Дан номер месяца (1 – январь, 2 – февраль, ...). Вывести число дней в этом месяце для невисокосного года (т. е. в феврале 28 дней).</p> <p>3. Дано целое число в диапазоне от 0 до 9. Вывести строку – название соответствующей цифры на русском языке (0 – «ноль», 1 – «один», 2 – «два», ...).</p> <p>4. Дано целое число в диапазоне от 1 до 5. Вывести строку – словесное описание соответствующей оценки (1 – «плохо», 2 – «неудовлетворительно», 3 – «удовлетворительно», 4 – «хорошо», 5 – «отлично»).</p> <p>5. Арифметические действия над числами пронумерованы следующим образом: 1 – сложение, 2 – вычитание, 3 – умножение, 4 – деление. Дан номер действия и два числа А и В (В не равно нулю). Выполнить над числами указанное действие и вывести результат.</p> <p>...</p> <p>Задание 2. Разработка блок-схемы процесса.</p> <p>На основе сведений о блок-схемах алгоритмов из предыдущей лабораторной работы, составить описание последовательности действий для абстрактного Исполнителя. Обязательно использовать подпрограммы для повторяющихся действий.</p> <p>Варианты:</p> <p>1 – Ловля рыбы.</p> <p>2 – Совершение покупок в магазине.</p> <p>3 – Приготовление пищи.</p> <p>4 – Чтение новостей в Internet.</p> <p>5 – Просмотр телевизора.</p> <p>...</p>	
Лабораторная работа № 2.1	Выполнение лабораторной работы.	<p>Вычисления и расчёты с целыми типами данных..</p> <p>Задание 1. Создать консольную программу для решения задачи в соответствии с вариантом. Значения переменных для вычислений должны вводиться с клавиатуры (например, посредством операторов Write('n='); Readln(n);), а результат выводиться в консольную строку оператором Writeln(<результат>).</p> <p>Варианты:</p> <p>1. Дано целое число N ($0 < N < 1000$), определяющее возраст черепахи в годах. В</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>зависимости от введённого значения грамотно сформировать фразу: «Черепaxe <N> (год/года/лет)».</p> <p>2. Дано целое число N ($0 < N < 1000$), определяющее количество пойманных рыб. В зависимости от введённого значения грамотно сформировать фразу: «Поймано <N> (рыб/рыбы/рыба)».</p> <p>3. Дано целое число N ($0 < N < 1000$), определяющее количество найденных грибов. В зависимости от введённого значения грамотно сформировать фразу: «Найдено <N> (грибов/гриба/гриб)».</p> <p>4. Дано целое число N ($0 < N < 1000$), определяющее количество учеников в классе. В зависимости от введённого значения грамотно сформировать фразу: «В классе <N> (ученик/ученика/учеников)».</p> <p>5. Дано целое число N ($0 < N < 1000$), определяющее количество участников соревнований. В зависимости от введённого значения грамотно сформировать фразу: «Соревновалось <N> (участник/участника/участников)».</p> <p>...</p> <p>Задание 2. Создать приложение с графическим интерфейсом для решения задачи в соответствии с вариантом. Значения переменных для вычислений должны вводиться в редактируемых элементах типа TEdit, а результат выводиться в текстовое поле типа TMemo.</p> <p>В заголовок окна приложения записать свои данные: фамилию, имя и номер варианта. Варианты:</p> <p>1. Дано целое число N. Посчитать и вывести количество и сумму его цифр.</p> <p>2. Возвести целое число N в квадрат без операции умножения, используя свойство: квадрат числа N равен сумме N первых нечетных чисел. Вывести все нечетные числа, используемые для вычислений.</p> <p>3. Разложить введенное с клавиатуры произвольное целое число N на все возможные целые простые множители. (Например: $20 = 2 \cdot 2 \cdot 5$; $21 = 3 \cdot 7$; $22 = 2 \cdot 11$; $23 = 1 \cdot 23$).</p> <p>4. Для введенного с клавиатуры произвольного целого числа N определить и вывести все возможные целые делители, кроме 1 и самого числа. Если число не имеет таких делителей, сообщить об этом. (Например: для 20 – 2,4,5,10; для 21 – 3,7; для 23 – «нет делителей»).</p> <p>5. Даны целые числа N и M. Найти на интервале <от меньшего из них до</p>	

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>большого из них > все целые числа, у которых результат деления на 2 является целым четным числом. Вывести все эти числа.</p> <p>...</p>	
Лабораторная работа № 2.2	Выполнение лабораторной работы.	<p>Вычисления и расчёты с вещественными типами данных..</p> <p>Для всех вариантов и заданий: создать приложение с графическим интерфейсом для решения задачи в соответствии с вариантом задания. Значения исходных данных (переменных) для вычислений должны вводиться в редактируемых элементах типа TEdit.Text.</p> <p>Варианты:</p> <ol style="list-style-type: none"> 1. Даны три вещественных числа x, y, z. Определить, существует ли треугольник с длинами сторон x, y, z. Если треугольник существует, определить его тип (равносторонний, равнобедренный, прямоугольный) и вычислить все углы в градусах. Напечатать длины сторон с противолежащими углами. 2. Даны три вещественных числа x, y, z. Определить, существует ли треугольник с длинами сторон x, y, z. Если треугольник существует, вычислить для него радиусы вписанной и описанной окружностей. 3. Даны два отрезка A и B, заданные координатами их концов $(AX1, AY1), (AX2, AY2)$ и $(BX1, BY1), (BX2, BY2)$ – целыми числами. Определить, пересекаются ли они, в случае пересечения вычислить координаты (вещественные числа) точки их пересечения. 4. Отрезок, заданный координатами концов $(X1, Y1), (X2, Y2)$ – целыми числами, определяет прямую, делящую пространство на две полуплоскости. Даны две точки A и B, заданные координатами (AX, AY) и (BX, BY) – целыми числами. Определить, расположены ли точки в одной полуплоскости, т.е. по одну сторону от прямой. 5. Луч, заданный координатами точки (X, Y) и углом ALPHA в градусах – целыми числами, определяет прямую, делящую пространство на две полуплоскости. Даны две точки A и B, заданные координатами (AX, AY) и (BX, BY) – целыми числами. Определить, расположены ли точки в одной полуплоскости, т.е. по одну сторону от прямой. <p>...</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2
Лабораторная работа № 3.1	Выполнение лабораторной работы.	<p>Работа с символьными данными.</p> <p>Создать приложение с графическим интерфейсом для решения задачи в</p>	ОПК-3: ИД-ОПК-3.1

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>соответствии с вариантом задания. Исходные строки должны вводиться в редактируемом строковом поле типа TМето. В каждой строке имеются группы символов («подстроки»), которые разделяются одним или несколькими пробелами. Пробелы могут находиться в начале и в конце строки. Подстрока или группа символов – это последовательность букв, цифр или знаков, отделенная пробелами от других групп, или находящаяся в начале/конце строки.</p> <p>В результирующее другое поле типа TМето должны быть перенесены (и/или преобразованы) только те строки, которые соответствуют условиям в вариантах заданий.</p> <p>Варианты:</p> <ol style="list-style-type: none"> 1. Симметричные группы символов из строки, если они состоят только из цифр. Например, строка: ‘ !2! 12321 reter 123 508805 lw1’; результат: ‘12321 508805’ 2. Симметричные группы символов из строки, если они состоят только из латинских букв. Например, строка: ‘ alla abba 787 typepyt lqq1’; результат: ‘abba typepyt’; 3. Группы символов, состоящие только из латинских букв, если символы расположены в алфавитном порядке по возрастанию. Например, строка: ‘ 12*er faw acf ab2c fghjk’; результат: ‘acf fghjk’; 4. Группы символов, состоящие только из латинских букв, если символы расположены в алфавитном порядке по убыванию. Например, строка: ‘ 12*er yfa y1fa acf ab2c ca’; результат: ‘yfa ca’; 5. Группы символов, состоящие только из цифр, если цифры расположены по возрастанию. Например, строка: ‘ 1276 357 12e6 0279 ’; результат: ‘357 0279’; <p>...</p>	ИД-ОПК-3.2
Лабораторная работа № 3.2	Выполнение лабораторной работы.	<p>Работа с матрицами (двумерными массивами).</p> <p>Создать приложение с графическим интерфейсом, производящее вычисления и обработку данных в соответствии с вариантом задания.</p> <p>Размерность обрабатываемой квадратной матрицы должна быть задана в редактируемом поле TEdit и не превышать количество зарезервированной размерности массива.</p> <p>Исходные данные должны вводиться в двумерный массив и отображаться в</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>редактируемой таблице типа TStringGrid. Они могут вводиться ручным способом или через датчик случайных чисел на интервале <-100..100>.</p> <p>Введённые данные должны быть обработаны согласно варианту задания. Преобразованный двумерный массив должен быть выведен в итоговую таблицу типа TStringGrid. Если результат является одномерным массивом (вектором), то он может быть выведен в текстовое поле типа TMemo. Скалярные (имеющие одно значение) результаты расчётов и вычислений могут быть выведены в виде текстовых ответов TLabel.</p> <p>Если результатами вычислений являются вещественные числа, то выводить их с точностью до 1-го знака после точки.</p> <p>Варианты:</p> <ol style="list-style-type: none"> 1. Вычислить среднее арифметическое элементов каждого из четных столбцов исходной матрицы. 2. Поменять местами строку, содержащую максимальный элемент, со строкой, содержащей минимальный элемент. Если оба таких элемента находятся в одной строке, то вывести сообщение и номер строки. 3. Получить новую матрицу вещественного типа путем деления всех элементов исходной матрицы на ее наименьший по модулю элемент (если он не равен 0). 4. Получить новую матрицу вещественного типа путем умножения всех элементов исходной матрицы на ее наибольший по модулю элемент (если он не равен 0). 5. Вычислить среднее арифметическое модулей элементов каждого из нечетных столбцов матрицы. <p>...</p>	
Лабораторная работа № 3.3	Выполнение лабораторной работы.	<p>Работа с записями.</p> <p>Создать приложение с графическим интерфейсом, производящее вычисления и обработку данных в соответствии с вариантом задания.</p> <p>Исходные данные должны вводиться посредством формы ввода, в которой используются поля типа TEdit и TComboBox. Результат ввода отображается в таблице типа TStringGrid.</p> <p>Промежуточные результаты выборки могут быть выведены в таблицу типа TStringGrid или в текстовое поле типа TMemo. Итоговый результат расчётов и вычислений может быть выведен в виде текстового ответа TLabel или добавлен в</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

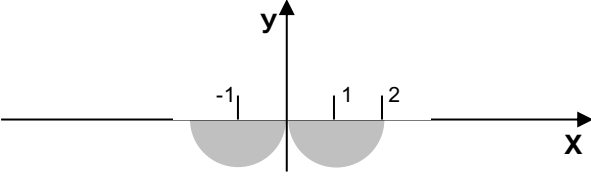
№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>текстовое поле типа ТМето.</p> <p>Варианты:</p> <ol style="list-style-type: none"> 1. Создать запись «Покупатель» с полями: «Имя» (строка), «Пол» (символ), «Цена товара» (вещественное число), «Количество товаров» (целое число). Имена покупателей могут повторяться, т.е. один покупатель может покупать разные товары в разном количестве. Вывести информацию о покупателе-мужчине, купившем больше всего товаров. 2. Создать запись «Покупатель» с полями: «Имя» (строка), «Пол» (символ), «Цена товара» (вещественное число), «Количество товаров» (целое число). Имена покупателей могут повторяться, т.е. один покупатель может покупать разные товары в разном количестве. Вывести информацию о покупателе-женщине, потратившей больше всего денег. 3. Создать запись «Покупатель» с полями: «Имя» (строка), «Пол» (символ), «Цена товара» (вещественное число), «Количество товаров» (целое число). Имена покупателей могут повторяться, т.е. один покупатель может покупать разные товары в разном количестве. Вывести информацию о покупателе-женщине, купившей больше всего товаров. 4. Создать запись «Покупатель» с полями: «Имя» (строка), «Пол» (символ), «Цена товара» (вещественное число), «Количество товаров» (целое число). Имена покупателей могут повторяться, т.е. один покупатель может покупать разные товары в разном количестве. Вывести информацию о покупателе-мужчине, потратившем больше всего денег. 5. Создать запись «Покупатель» с полями: «Имя» (строка), «Пол» (символ), «Цена товара» (вещественное число), «Количество товаров» (целое число). Имена покупателей могут повторяться, т.е. один покупатель может покупать разные товары в разном количестве. Вывести информацию о покупателе-мужчине, потратившем меньше всего денег. <p>...</p>	

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
Лабораторная работа № 4.1	Выполнение лабораторной работы.	<p>Использование подпрограмм и модулей.</p> <p>Создать приложение с графическим интерфейсом и двумя визуальными формами, производящее обработку данных одномерного массива в соответствии с вариантом задания.</p> <p>Значения элементов массива должны вводиться в одной форме, а все операции вывода производиться во второй форме. Данные должны обрабатываться в отдельно созданной подпрограмме (процедуре или функции согласно варианту задания). Массив и количество элементов массива должны передаваться в подпрограмму в качестве параметров.</p> <p>Исходные данные должны вводиться посредством формы ввода, в которой используется поле типа TEdit. Исходный массив должен отображаться в текстовом поле типа TMemo или в таблице типа TStringGrid. Если в результате вычислений элементы массива изменяются, то полученный массив также необходимо вывести в текстовом поле типа TMemo или в таблице типа TStringGrid. Итоговый результат расчётов и вычислений может быть выведен в виде текстового ответа TLabel или добавлен в текстовое поле типа TMemo.</p> <p>Варианты:</p> <ol style="list-style-type: none"> 1. Массив целых чисел. В процедуре отсортировать массив по возрастанию. 2. Массив целых чисел. В процедуре отсортировать массив по убыванию. 3. Массив вещественных чисел. В процедуре отсортировать массив по возрастанию. 4. Массив вещественных чисел. В процедуре отсортировать массив по убыванию. 5. Массив целых чисел. В процедуре удалить из массива четные элементы. <p>Размерность итогового массива должна уменьшиться.</p> <p>...</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2
Лабораторная работа № 4.2	Выполнение лабораторной работы.	<p>Использование библиотек DLL.</p> <p>Создать приложение с графическим интерфейсом, производящее вычисления и обработку данных в подпрограммах, находящихся в библиотеках DLL, в соответствии с вариантом задания. Исходные данные должны вводиться в полях типа TEdit. Результаты расчётов и вычислений могут быть выведены в текстовое поле типа TMemo.</p> <p>Исходными данными являются два целых числа. С ними необходимо произвести вычисления в соответствии с вариантом задания.</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
Лабораторная работа № 5.1	Выполнение лабораторной работы.	<p>Линейное программирование.</p> <p>Вариант 1</p> <ol style="list-style-type: none"> 1) Написать программу пересчёта расстояния из вёрст в километры (1 верста = 1066.8 м). Программа должна обеспечивать ввод расстояния в верстах и вывод расстояния в километрах. 2) Написать программу для расчёта по двум формулам (результат вычисления по первой формуле должен совпадать со второй). $z_1 = \frac{\sin 2\alpha + \sin 5\alpha - \sin 3\alpha}{\cos \alpha + 1 - 2\sin^2 2\alpha}; \quad z_2 = 2 \sin \alpha$ <p>Вариант 2</p> <ol style="list-style-type: none"> 1) Написать программу пересчёта веса из фунтов в килограммы (1 фунт = 405.9 грамм). Программа должна обеспечивать ввод веса в фунтах и вывод веса в килограммах. 2) Написать программу для расчёта по двум формулам (результат вычисления по первой формуле должен совпадать со второй). $z_1 = \frac{1 - 2\sin^2 \alpha}{1 + \sin 2\alpha}; \quad z_2 = \frac{1 - \operatorname{tg} \alpha}{1 + \operatorname{tg} \alpha}$ <p>Вариант 3</p> <ol style="list-style-type: none"> 1) Написать программу вычисления стоимости поездки на автомобиле на дачу (туда и обратно). Программа должна позволять вводить следующие исходные данные: расстояние до дачи (км); расход бензина (количество литров на 100 км пробега); цена одного литра бензина (руб.). 2) Написать программу для расчёта по двум формулам (результат вычисления по первой формуле должен совпадать со второй). 	<p>ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2</p>

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		$z_1 = \frac{x^2 + 2x - 3 + (x+1)\sqrt{x^2 - 9}}{x^2 - 2x - 3 + (x-1)\sqrt{x^2 - 9}}; \quad z_2 = \sqrt{\frac{x+3}{x-3}}$ <p>Вариант 4</p> <ol style="list-style-type: none"> 1) Написать программу вычисления силы тока в электрической цепи. Программа должна позволять вводить величину напряжения (В) и сопротивления (Ом). 2) Написать программу для расчёта по двум формулам (результат вычисления по первой формуле должен совпадать со второй). $z_1 = \cos^2\left(\frac{3}{8}\pi - \frac{\alpha}{4}\right) - \cos^2\left(\frac{11}{8}\pi + \frac{\alpha}{4}\right); \quad z_2 = \frac{\sqrt{2}}{2} \sin \frac{\alpha}{2}$ <p>Вариант 5</p> <ol style="list-style-type: none"> 1) Написать программу пересчёта расстояния из дюймов в сантиметры (1 дюйм = 0.0254 м). Программа должна обеспечивать ввод расстояния в дюймах и вывод расстояния в сантиметрах. 2) Написать программу для расчёта по двум формулам (результат вычисления по первой формуле должен совпадать со второй). $z_1 = 2 \sin^2(3\pi - 2\alpha) \cos^2(5\pi + 2\alpha); \quad z_2 = \frac{1}{4} - \frac{1}{4} \sin\left(\frac{5}{2}\pi - 8\alpha\right)$ <p>...</p>	
Лабораторная работа № 6.1	Выполнение лабораторной работы.	<p>Расчетные задачи.</p> <p>Вариант 1. Подсчитать k-количество цифр в десятичной записи целого неотрицательного числа n.</p> <p>Вариант 2. Дано n вещественных чисел. Вычислить разность между максимальным и минимальным из них.</p> <p>Вариант 3. Дана непустая последовательность различных натуральных чисел, за которой следует 0. Определить порядковый номер наименьшего из них.</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>Вариант 4. Даны целое $n > 0$ и последовательность из n вещественных чисел, среди которых есть хотя бы одно отрицательное число. Найти величину наибольшего среди отрицательных чисел этой последовательности.</p> <p>Вариант 5. Дано n вещественных чисел. Определить, образуют ли они возрастающую последовательность.</p> <p>...</p>	
Лабораторная работа № 7.1	Выполнение лабораторной работы.	<p>Циклы.</p> <p>Часть 1. Единое задание для закрепления пройденного материала. Приложение реализовать в консольном и в визуальном исполнении.</p> <p>1) Используя цикл <code>for</code>, написать программу, которая выводит таблицу значений функции $y = 2x^2 - 5x - 8$. При этом пользователь должен ввести диапазон и шаг изменения аргумента x.</p> <p>2) Используя цикл <code>while</code>, написать программу, которая определяет произведение цифр введённого пользователем натурального числа типа <code>int</code>. Предусмотреть обработку ошибок.</p> <p>Часть 2. Индивидуальное задание по вариантам.</p> <p>Вариант 1. Вычислить дробную часть среднего геометрического трех заданных положительных чисел.</p> <p>Вариант 2. Вычислить длину окружности, площадь круга и объем шара одного и того же радиуса.</p> <p>Вариант 3. Вычислить периметр и площадь прямоугольного треугольника по длинам двух катетов.</p> <p>Вариант 4. По координатам трех вершин некоторого треугольника найти его площадь и периметр.</p> <p>Вариант 5. По длинам двух сторон некоторого треугольника и углу между ними найти длину третьей стороны и вычислить площадь этого треугольника.</p> <p>...</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2
Лабораторная работа № 8.1	Выполнение лабораторной работы.	<p>Ветвления и циклы.</p> <p>Часть 1. Единое задание для закрепления пройденного материала. Приложения реализовать в консольном и в визуальном исполнении.</p> <p>1) Дана заштрихованная область (см. рис.). Написать программу, которая позволяет вводить с клавиатуры точку с координатами $(x; y)$ и определять</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>попадает ли данная точка в область или нет.</p>  <p>2) Написать программу, которая выводит в виде таблицы значения функции $y=f(x)$ на интервале $x \in [1;10]$ с шагом $dX = 1$:</p> $f(x) = \begin{cases} x^2 + 4x - 10, & \text{если } x=5 \text{ или } x=7 \\ \sqrt[2]{4x} + \sqrt[3]{16x}, & \text{если } x=4 \\ x^3 - 2, & \text{в остальных случаях} \end{cases}$ <p>Необходимо реализовать программу в двух версиях. В одной версии следует применить циклическую конструкцию с условиями, а в другой версии – циклическую конструкцию со switch.</p> <p>Часть 2. Индивидуальное задание по вариантам. Приложение реализовать в консольном и в визуальном исполнении.</p> <p>Вариант 1. Вычислить значение функции</p> $f(X) = \left[\begin{array}{l} X^2 + 12 * X, \text{если } X = 3 \\ 3 * X^{3.5} + 0.78 * X, \text{если } X = 13 \\ 12 * X + 35, \text{в другом варианте } X \end{array} \right] \quad X \in [0,32], \Delta X = 1;$ <p>Написать два варианта программы - с использованием оператора if и с использованием оператора switch. Результат представить в виде таблицы.</p>	

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>Вариант 2. Вычислить значение функции</p> $f(X) = \begin{cases} (X+12)^{3+x} + X, & \text{если } X = 6 \\ 3 * X^3 + 0.78 * X, & \text{если } X = 2 \\ 0.56 * X^{X/24}, & \text{в другом варианте } X \end{cases} \quad X \in [0, 10], \Delta X =$ <p>Написать два варианта программы - с использованием оператора if и с использованием оператора switch. Результат представить в виде таблицы.</p> <p>Вариант 3. Вычислить значение функции</p> $f(X) = \begin{cases} (25 + X) / (X^2 + 24), & \text{если } X = 5 \text{ и } X = 3 \\ 3 * X^{3.5} + 0.78 * X, & \text{если } X = 10 \\ X^{X*2}, & \text{в другом варианте } X \end{cases} \quad X \in [-1, 20], \Delta X = 1;$ <p>Написать два варианта программы - с использованием оператора if и с использованием оператора switch. Результат представить в виде таблицы.</p> <p>Вариант 4. Вычислить значение функции</p> $f(X) = \begin{cases} (25 + X) / (23 * X + 24), & \text{если } X = 3 \text{ и } X = 0 \\ X^{3.5} + 0.78 * \sqrt[3]{2 * X}, & \text{если } X = 10 \\ X^5, & \text{в другом варианте } X \end{cases} \quad X \in [0, 20], \Delta X = 1;$ <p>Написать два варианта программы - с использованием оператора if и с</p>	

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>использованием оператора switch. Результат представить в виде таблицы.</p> <p>Вариант 5. Вычислить значение функции</p> $f(X) = \begin{cases} \sqrt[5]{X+25}, & \text{если } X=3 \text{ и } X=0 \\ \sqrt{X} + 0.78 * \sqrt[3]{2*X}, & \text{если } X=10 \\ X^X, & \text{в другом варианте } X \end{cases} \quad X \in [-1, 20], \Delta X = 1;$ <p>Написать два варианта программы - с использованием оператора if и с использованием оператора switch. Результат представить в виде таблицы.</p> <p>...</p>	
Лабораторная работа № 9.1	Выполнение лабораторной работы.	<p>Массивы.</p> <p>Часть 1. Единое задание для закрепления пройденного материала. Приложения реализовать в консольном исполнении.</p> <p>1) Написать программу, которая позволяет вводить одномерный целочисленный массив из 9 элементов и вычислять: а) сумму положительных элементов массива с чётными номерами; б) количество элементов массива, равных нулю; в) вещественный результат деления максимального элемента массива на минимальный. В случае возникновения исключительных ситуаций программа должна выводить соответствующие сообщения.</p> <p>2) Написать программу, которая позволяет вводить целочисленную матрицу 3×4, выводить её в наглядном формате и определять номер столбца, сумма элементов которого минимальна. Программа должна позволять производить вычисления столько раз, сколько угодно пользователю.</p> <p>Часть 2. Индивидуальное задание по вариантам. Приложение реализовать в консольном исполнении.</p> <p>Вариант 1. Дан массив целых чисел. Написать программу для сортировки</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>массива по возрастанию.</p> <p>Вариант 2. Дан массив целых чисел. Написать программу для определения максимальной и минимальной суммы двух соседних элементов массива.</p> <p>Вариант 3. Дан массив целых чисел. Написать программу для определения максимальной и минимальной разницы между двумя соседними элементами массива.</p> <p>Вариант 4. Дан массив вещественных чисел. Написать программу для сортировки массива по возрастанию целой части его элементов.</p> <p>Вариант 5. Дан массив целых чисел. Написать программу для сортировки заданного массива по убыванию суммы цифр элементов.</p> <p>...</p>	
Лабораторная работа № 9.2	Выполнение лабораторной работы.	<p>Строки типа char.</p> <p>Часть 1. Единое задание для закрепления пройденного материала. Приложения реализовать в консольном и в визуальном исполнении.</p> <p>1) Написать программу, которая позволяет вводить с клавиатуры строку, содержащую более одного слова, и определять: а) количество символов введённой строки; б) содержит ли строка слова одинаковой длины; в) количество запятых. Необходимо учесть, что при вводе строки, могут использоваться не только пробелы, но и знаки пунктуации.</p> <p>2) Напишите программу, которая позволяет вводить с клавиатуры строку, содержащую более одного слова, и определять количество слов, заканчивающихся на заданную пользователем букву. Необходимо учесть, что при вводе строки, могут использоваться не только пробелы, но и знаки пунктуации</p> <p>Часть 2. Индивидуальное задание по вариантам. Приложение реализовать в консольном и/или в визуальном исполнении.</p> <p>Вариант 1. В заданном тексте удалить слова, начинающиеся с заданной буквы. При написании программы нельзя пользоваться стандартными функциями обработки строк.</p> <p>Вариант 2. Вывести на экран заданный текст, удалив из него лишние пробелы, т.е. из нескольких подряд идущих пробелов оставить только один. При написании</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

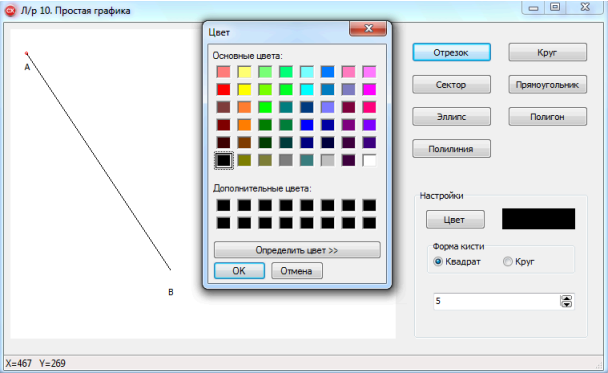
№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>программы нельзя пользоваться стандартными функциями обработки строк.</p> <p>Вариант 3. Задан текст, заканчивающийся точкой. Вывести на экран сначала все цифры, входящие в него, а затем все остальные литеры. При написании программы нельзя пользоваться стандартными функциями обработки строк.</p> <p>Вариант 4. Определить, сколько различных литер входит в заданный текст, заканчивающийся точкой. При написании программы нельзя пользоваться стандартными функциями обработки строк.</p> <p>Вариант 5. Заданный текст вывести на экран по строкам, понимая под строкой либо очередные 60 символов, если среди них нет запятой, либо текст до запятой включительно. При написании программы нельзя пользоваться стандартными функциями обработки строк.</p> <p>...</p>	
Лабораторная работа № 10.1	Выполнение лабораторной работы.	<p>Функции.</p> <p>Часть 1. Единое задание для закрепления пройденного материала. Приложения реализовать в визуальном исполнении.</p> <ol style="list-style-type: none"> 1) Написать функцию, которая возвращает возведённое в степень число, и программу, использующую эту функцию для возведения введённого пользователем числа в заданную им степень. 2) Написать программу, позволяющую сортировать введенный пользователем целочисленный одномерный массив по убыванию или по возрастанию (в зависимости от выбора пользователя). Сортировку по убыванию необходимо реализовать методом пузырька, а сортировку по возрастанию – методом выбора. Указанные виды сортировок нужно оформить в виде отдельных функций. 3) Напишите функцию, которая переворачивает строку (массив типа <code>char</code>). Используйте цикл <code>for</code>, который меняет местами первый и последний символы, затем следующие и т.д. Строка должна передаваться в функцию как аргумент. Напишите программу, которая должна принимать от пользователя строку (содержащую более одного слова), вызывать функцию, а затем 	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

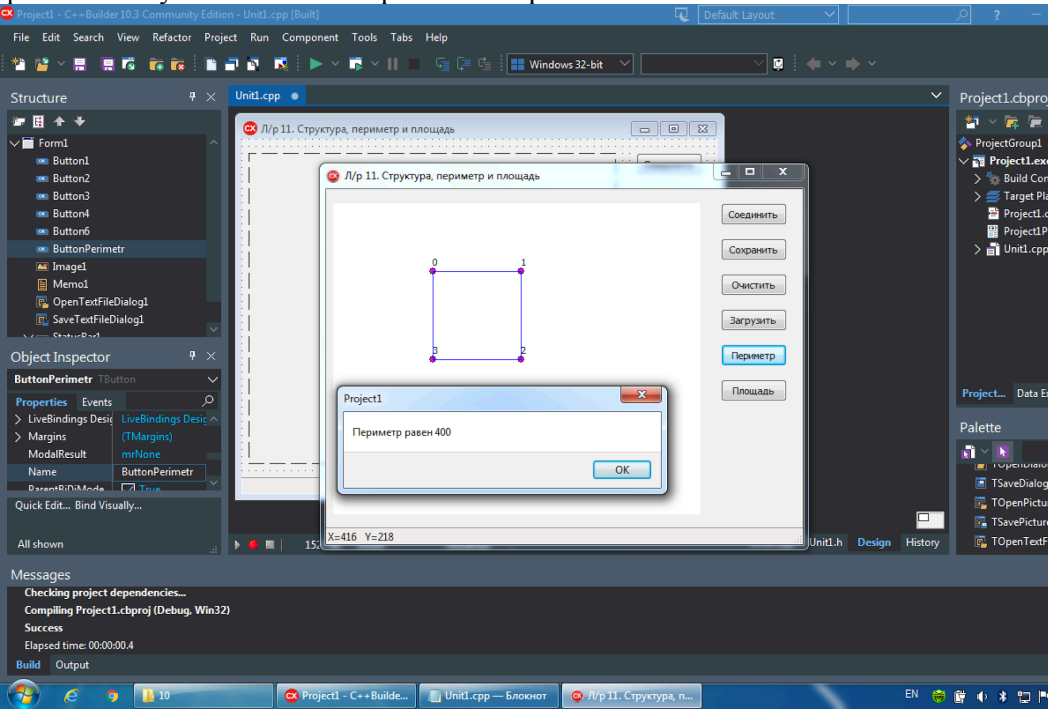
№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>выводить полученный результат.</p> <p>4) Напишите функцию вычисления корней квадратного уравнения. Программа должна принимать от пользователя коэффициенты, вызывать функцию, а затем выводить полученный результат. Параметрами функции должны быть коэффициенты и корни уравнения. Такие аргументы как корни уравнения необходимо передавать по ссылке. Значение, возвращаемое функцией, должно передавать в вызывающую программу информацию о наличии корней: 2 – два разных корня; 1 – корни одинаковые; 0 – уравнение не имеет решения. Кроме того, функция должна проверять корректность исходных данных. Если исходные данные неверные, то функция должна возвращать -1.</p> <p>Часть 2. Индивидуальное задание по вариантам. Приложение реализовать в консольном и/или в визуальном исполнении.</p> <p>Вариант 1. Дана строка. Написать функцию для определения, входит ли в строку заданное слово, или нет. При написании программы необходимо использовать стандартные функции обработки строк.</p> <p>Вариант 2. Дана строка символов. Написать функцию для определения одинаковы или нет второе и последнее слово в этой строке. При написании программы необходимо использовать стандартные функции обработки строк.</p> <p>Вариант 3. Дана строка символов. Написать функцию для определения слов, которые начинаются и заканчиваются на одинаковые буквы. При написании программы необходимо использовать стандартные функции обработки строк.</p> <p>Вариант 4. Дана строка символов. Написать функцию для определения, входит ли в эту строку заданное слово. Если входит, то его нужно удалить. При написании программы необходимо использовать стандартные функции обработки строк.</p> <p>Вариант 5. Дана строка символов. Написать функцию для формирования строки с обратным порядком слов. При написании программы необходимо использовать стандартные функции обработки строк.</p> <p>...</p>	
Лабораторная работа № 11.1	Выполнение лабораторной работы.	Указатели и функции. Динамическое выделение памяти. Часть 1. Единое задание для закрепления пройденного материала. Приложения реализовать в консольном исполнении.	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>1) Напишите программу, которая позволяет вводить динамический одномерный целочисленный массив и посредством вызова функции определять минимальную сумму между двумя соседними элементами массива. При передаче массива в функцию используйте указатели, а размерность передавайте по ссылке. Применение глобальных переменных категорически запрещено.</p> <p>2) Напишите программу, позволяющую вводить динамический многомерный целочисленный массив и посредством функций осуществлять: 1) вывод введённого массива на экран в наглядном формате; 2) вычисление суммы элементов в тех строках, которые содержат хотя бы один отрицательный элемент; 3) определение минимального из чисел, встречающихся в заданной матрице более одного раза. При передаче аргументов в функции старайтесь использовать указатели. Применение глобальных переменных категорически запрещено</p> <p>Часть 2. Индивидуальное задание по вариантам. Приложение реализовать в консольном исполнении. При передаче массива в функцию используйте указатели, а размерность передавайте по ссылке. Применение глобальных переменных категорически запрещено.</p> <p>Вариант 1. Пусть пользователь вводит динамическую матрицу размером $N \times M$. Написать функцию для вычисления суммы диагональных элементов.</p> <p>Вариант 2. Пусть пользователь вводит динамическую матрицу размером $N \times M$. Написать функцию для вычисления максимальных и минимальных элементов в столбцах.</p> <p>Вариант 3. Пусть пользователь вводит динамическую матрицу размером $N \times M$. Написать функцию для вычисления максимальной и минимальной суммы элементов в строках.</p> <p>Вариант 4. Пусть пользователь вводит динамическую матрицу размером $N \times M$. Написать функцию для сортировки строк матрицы по убыванию.</p> <p>Вариант 5. Пусть пользователь вводит динамическую матрицу размером $N \times M$. С помощью функции определить максимальный элемент.</p> <p>...</p>	

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
Лабораторная работа № 12.1	Выполнение лабораторной работы.	<p>Потоковые классы C++ и файлы. Текстовые и бинарные файлы.</p> <p>Часть 1. Единое задание для закрепления пройденного материала. Приложение 1 реализовать в консольном и в визуальном исполнении, приложение 2 – в консольном исполнении.</p> <p>1) Напишите программу, которая позволяет: а) выводить на экран содержимое указанного текстового файла; б) записывать в другой текстовый файл только те предложения, которые содержат введённое с клавиатуры слово. Если указанное слово отсутствует в тексте, программа должна выводить сообщение и не должна создавать выходной файл. По возможности старайтесь использовать динамическое выделение памяти.</p> <p>2) Напишите программу, которая позволяет создавать список данных железнодорожных поездов и хранить его в бинарном файле (например, ...\\railwaybase.dat). Каждая запись бинарного файла содержит: номер поезда (sizeof(int) байт); пункт отправления-прибытия (30 байт); количество проданных билетов (sizeof(int) байт); цена одного билета (sizeof(float) байт). Программа должна позволять: а) добавлять в двоичный файл записи, причём столько раз, сколько угодно пользователю; б) показывать все записи в наглядном формате, причём стоимость одного билета должна выводиться с двумя знаками после запятой; в) подсчитывать выручку указанного номера поезда. Пользователь должен иметь возможность в любой момент выбрать необходимое ему действие с помощью меню, например следующего вида:</p> <pre> =====RailwayBase===== 1 - dobavit zapis; 2 - pokazat vse zapisi; 3 - pokazat vyruчку poezda; 4 - vyhod; Vash vybor: </pre> <p>Меню реализуйте с помощью функции, возвращающей значение.</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>Предусмотрите обработку исключительных ситуаций. Часть 2. Индивидуальное задание по вариантам. Приложение реализовать в консольном исполнении. Вариант 1. Дан текстовый файл, состоящий из 5 строк. Написать функцию для сравнения первого слова второй строки и последнего слова пятой строки. Вариант 2. Дан текстовый файл, состоящий из некоторого количества строк. Написать функцию для сравнения первой и последней строки. Вариант 3. Дан текстовый файл, состоящий из некоторого количества строк. Написать функцию для создания нового файла с обратным порядком строк. Вариант 4. Дан текстовый файл, состоящий из некоторого количества строк. Написать функцию для создания нового файла, содержащего вторые слова строк первого файла. Вариант 5. Даны два текстовых файла, состоящие из некоторого количества строк. Написать функцию для сравнения этих файлов. ...</p>	
Лабораторная работа № 13.1	Выполнение лабораторной работы.	<p>Массивы и функции. Вариант 1. Преобразовать одномерный целочисленный массив таким образом, чтобы сначала располагались все положительные элементы, а потом – все отрицательные (элементы, равные 0, считать положительными). Вариант 2. Сжать одномерный вещественный массив, удалив из него все элементы, модуль которых не превышает 1. Освободившиеся в конце массива элементы заполнить нулями. Вариант 3. Сжать одномерный вещественный массив, удалив из него все элементы, модуль которых находится в интервале [a,b]. Освободившиеся в конце массива элементы заполнить нулями. Вариант 4. Преобразовать одномерный вещественный массив таким образом, чтобы сначала располагались все элементы, равные нулю, а потом – все остальные. Вариант 5. Преобразовать одномерный целочисленный массив таким образом,</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине – элементы, стоявшие в четных позициях.</p> <p>...</p>	
Лабораторная работа № 13.2	Выполнение лабораторной работы.	<p>Простая графика. Работа с компонентом Image. Построение графических примитивов: отрезок, круг, сектор, эллипс, прямоугольник, полигон, полилиния. Рисование на холсте с динамической закраской. Выбор цвета, толщины и формы кисти.</p>  <p>Вариант 1. Воспроизвести, реализовав ластик по нажатию правой кнопки мышки. Вариант 2. Воспроизвести, добавив форму кисти «ромб». Вариант 3. Воспроизвести, добавив форму кисти «пентагон». Вариант 4. Воспроизвести, добавив форму кисти «треугольник». Вариант 5. Воспроизвести, добавив форму кисти «эллипс».</p> <p>...</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2
Лабораторная работа № 13.3	Выполнение лабораторной работы.	<p>Часть 1. Единое задание для закрепления пройденного материала. Работа с графикой с применением структуры фигур:</p> <pre> struct Koordinata { int x; int y; }; </pre> <p>Построение многоугольников по точкам, сохранение их в текстовый файл,</p>	ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		<p>очистка холста, загрузка данных из файла и восстановление нарисованных многоугольников. Расчет периметра и площади нарисованных и загруженных из файла многоугольников. Воспроизвести приложение.</p>  <p>Часть 2. Индивидуальное задание по вариантам. Приложение реализовать в консольном и/или в визуальном исполнении.</p> <p>Вариант 1. Нарисовать график функции $f(X) = 2 * X^3 + 3 * X^2$.</p> <p>Вариант 2. Нарисовать график функции $f(X) = X^2 + 25 * X$.</p> <p>Вариант 3. Нарисовать график функции $f(X) = \sin(X) + \sqrt{ X } * \cos(23 * X)$.</p> <p>Вариант 4. Нарисовать график функции $f(X) = \sin(X)$.</p> <p>Вариант 5. Нарисовать график функции $f(X) = \cos(X)$.</p>	

№ пп	Формы текущего контроля	Примеры типовых заданий	Формируемая компетенция
		...	

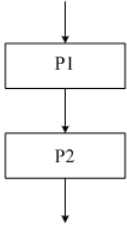
5.2. Критерии, шкалы оценивания текущего контроля успеваемости:

Наименование оценочного средства (контрольно-оценочного мероприятия)	Критерии оценивания	Шкалы оценивания	
		100-балльная система	Пятибалльная система
Лабораторная работа	Работа выполнена полностью. Нет ошибок в логических рассуждениях и в реализации задания в виде файла или выполняемой программы. Возможно наличие одной неточности или описки, не являющиеся следствием незнания или непонимания учебного материала и не влияющей на функциональные качества программы. Обучающийся показал полный объем знаний, умений в освоении пройденных тем и применение их на практике. Работа зачтена.		5
	Работа выполнена полностью, но выбран неэффективный алгоритм или метод реализации, обоснований шагов решения недостаточно. Допущена одна ошибка или два-три недочета, которые незначительно влияют на качество представленной работы. Работа зачтена.		4
	Допущены более одной ошибки или более двух-трех недочетов, которые оказывают значительное влияние на представляемый файл или компьютерную программу, ухудшают их информативность и функциональные возможности. Работа зачтена.		3
	Работа выполнена не полностью. Допущены грубые ошибки. Файлы не содержат необходимой информации, компьютерная программа выдаёт неправильные результаты при вычислении тестовых примеров. Работа не зачтена.		2
	Работа не выполнена.		

5.3. Промежуточная аттестация:

Форма промежуточной аттестации	Типовые контрольные задания и иные материалы для проведения промежуточной аттестации:	Формируемая компетенция
Экзамен:	1. Для временного хранения информации используется ...	ОПК-3:

Компьютерное тестирование	<p>a) принтер b) флеш-память c) оперативная память (ОЗУ) d) жесткий диск</p> <p>...</p> <p>6. Преобразование исходного текста программы, написанного на языке программирования, в язык, близкий к машинному называется...</p> <p>a) Генерацией b) Интерпретацией c) Компиляцией d) Исполнением</p> <p>7. Метод разработки задачи с помощью разбиения его на подзадачи называется:</p> <p>a) подчиненным b) нисходящим c) восходящим d) параллельным</p> <p>...</p> <p>11. Какая структура не относится к базовой при структурном подходе к созданию алгоритмов?</p> <p>a) Следование b) Изображение c) Ветвление d) Цикл</p> <p>12. Какое требование для алгоритма, адресуемого заданному исполнителю, является неверным?</p> <p>a) Дискретность (элементарность) b) Детерминированность (определённость, однозначность) c) Нерешительность (сомнительность) d) Результативность (конечность, сходимоссть)</p> <p>13. Какую структуру имеет алгоритм, представленный на рисунке?</p>	ИД-ОПК-3.1 ИД-ОПК-3.2
---------------------------	---	--------------------------

	 <p>a) разветвляющуюся b) циклическую с условием c) линейную d) циклическую с постусловием</p> <p>...</p>	
<p>Экзамен: в письменной форме по билетам</p>	<p>Билет № 1 Вопрос 1. Характеристика языка C++. В каких проектах он применяется? Принцип работы компилятора языка высокого уровня. Вопрос 2. Написать приложение на C++, которое позволяет вводить текст с клавиатуры и посредством функции удалять слова, начинающиеся с указанной буквы. Программу реализовать двумя способами: 1) использовать строки типа char или wchar_t и не пользоваться стандартными функциями обработки строк; 2) использовать строки типа string или вектор строк vector<string> и обязательно пользоваться стандартными функциями для string или vector<string>.</p> <p>Билет № 3 Вопрос 1. Основные особенности C++ и каковы различия между C и C++? Структура программы на языке C++. Директива препроцессора и заголовочные файлы. Что такое указатели в C++ и как их использовать? Вопрос 2. Разработать приложение на C++, которое позволяет вводить строку типа char (или wchar_t) и с помощью функции определять количество согласных букв в строке.</p> <p>Билет № 5 Вопрос 1. Предназначение функций стандартной библиотеки. Что представляет собой стандартная библиотека C++ и какие наиболее часто используемые функции в ней содержатся? Как происходит обработка исключений в C++? Вопрос 2. Создать приложение на C++, которое проверяет отсортирован ли считанный из текстового файла массив (или вектор) по возрастанию. Проверку организовать посредством</p>	<p>ОПК-3: ИД-ОПК-3.1 ИД-ОПК-3.2</p>

	<p>функции.</p> <p>Билет № 7 Вопрос 1. Класс памяти переменной в C++. Константы. Перечислите стандарты языка программирования C++. На что влияет выбранный разработчиком в опциях компиляции стандарт? Что нового они привнесли, приведите примеры. Вопрос 2. Написать программу на C++, которая получает на вход имена людей. Например, Boris, Vladimir, Anna, Petr. Необходимо вывести отсортированный по алфавиту список. Сортировку оформить с помощью отдельной функции.</p> <p>Билет № 9 Вопрос 1. Преобразование (приведение) типов данных. Для чего используются пространства имен в C++? Как работает оператор присваивания в C++ с точки зрения безопасности? Вопрос 2. Написать консольную программу на C++, позволяющую вводить два одномерных целочисленных массива. Необходимо с помощью функции найти наименьшие среди тех элементов первого массива, которые не входят во второй массив.</p> <p>...</p>	
--	---	--

5.4. Критерии, шкалы оценивания промежуточной аттестации учебной дисциплины:

Форма промежуточной аттестации	Критерии оценивания	Шкалы оценивания	
		100-балльная система	Пятибалльная система
Экзамен: компьютерное тестирование	За выполнение каждого тестового задания испытуемому выставляются баллы. За полностью правильный ответ к каждому заданию с выбором одного правильного варианта выставляется один балл, за неправильный — ноль. За задания с выбором нескольких правильных ответов или в заданиях с сопоставлениями испытуемый может получить менее 1 балла. Например, если правильных ответов в задании два, то за каждый он получает 0,5 балла, если правильных ответов три, то за каждый он получает 0,333 балла и т.п. Правила оценки всего теста:		5 85% - 100%
			4 65% - 84%
			3 41% - 64%
			2 40% и менее 40%

Форма промежуточной аттестации	Критерии оценивания	Шкалы оценивания	
Наименование оценочного средства		100-балльная система	Пятибалльная система
	вне зависимости от количества заданий в тесте общая сумма баллов за все правильные ответы пересчитывается тестирующей компьютерной системой в итоговые баллы. 10 итоговых баллов эквивалентны 100% правильных ответов. Для того, чтобы получить отличную, хорошую, удовлетворительную или неудовлетворительную оценки, итоговые баллы за промежуточные аттестации каждого семестра складываются с баллами за выполненные лабораторные работы.		
Экзамен: в письменной форме по билетам	<p>Обучающийся:</p> <ul style="list-style-type: none"> – демонстрирует знания, отличающиеся глубиной и содержательностью, дает полный исчерпывающий ответ, как на основные вопросы билета, так и на дополнительные; – свободно владеет научными понятиями, ведет диалог и вступает в научную дискуссию; – способен к интеграции знаний по определенной теме, структурированию ответа, к анализу положений существующих теорий, научных школ, направлений по вопросу билета; – логично и доказательно раскрывает проблему, предложенную в билете; – свободно выполняет практические задания повышенной сложности, предусмотренные программой, демонстрирует системную работу с основной и дополнительной литературой. <p>Ответ не содержит фактических ошибок и характеризуется глубиной, полнотой, уверенностью суждений, иллюстрируется примерами, в том числе из собственной практики.</p>		5
	<p>Обучающийся:</p> <ul style="list-style-type: none"> – показывает достаточное знание учебного материала, но допускает несущественные фактические ошибки, которые способен исправить самостоятельно, благодаря наводящему вопросу; – недостаточно раскрыта проблема по одному из вопросов билета; – недостаточно логично построено изложение вопроса; 		4

Форма промежуточной аттестации	Критерии оценивания	Шкалы оценивания	
Наименование оценочного средства		100-балльная система	Пятибалльная система
	<ul style="list-style-type: none"> – успешно выполняет предусмотренные в программе практические задания средней сложности, активно работает с основной литературой, – демонстрирует, в целом, системный подход к решению практических задач, к самостоятельному пополнению и обновлению знаний в ходе дальнейшей учебной работы и профессиональной деятельности. <p>В ответе раскрыто, в основном, содержание билета, имеются неточности при ответе на дополнительные вопросы.</p>		
	<p>Обучающийся:</p> <ul style="list-style-type: none"> – показывает знания фрагментарного характера, которые отличаются поверхностностью и малой содержательностью, допускает фактические грубые ошибки; – не может обосновать закономерности и принципы, объяснить факты, нарушена логика изложения, отсутствует осмысленность представляемого материала, представления о межпредметных связях слабые; – справляется с выполнением практических заданий, предусмотренных программой, знаком с основной литературой, рекомендованной программой, допускает погрешности и ошибки при теоретических ответах и в ходе практической работы. <p>Содержание билета раскрыто слабо, имеются неточности при ответе на основные и дополнительные вопросы билета, ответ носит репродуктивный характер. Неуверенно, с большими затруднениями решает практические задачи или не справляется с ними самостоятельно.</p>		3
	<p>Обучающийся обнаруживает существенные пробелы в знаниях основного учебного материала, допускает принципиальные ошибки в выполнении предусмотренных программой практических заданий.</p> <p>На большую часть дополнительных вопросов по содержанию экзамена затрудняется дать ответ или не дает верных ответов.</p>		2

5.5. Система оценивания результатов текущего контроля и промежуточной аттестации.

Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.

Форма контроля	100-балльная система	Пятибалльная система
Текущий контроль:		
Выполнение лабораторной работы		зачтено/не зачтено
Промежуточная аттестация экзамен		отлично хорошо
Итого за третий семестр (дисциплину) экзамен		удовлетворительно неудовлетворительно
Промежуточная аттестация экзамен		отлично хорошо
Итого за четвертый семестр (дисциплину) экзамен		удовлетворительно неудовлетворительно

Полученный совокупный результат конвертируется в пятибалльную систему оценок в соответствии с таблицей:

100-балльная система	пятибалльная система	
	зачет с оценкой/экзамен	зачет
85 – 100 баллов	отлично зачтено (отлично)	зачтено
65 – 84 баллов	хорошо зачтено (хорошо)	
41 – 64 баллов	удовлетворительно зачтено (удовлетворительно)	
0 – 40 баллов	неудовлетворительно	не зачтено

6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Реализация программы предусматривает использование в процессе обучения следующих образовательных технологий:

- проблемная лекция;
- проектная деятельность;
- групповые дискуссии;
- анализ ситуаций и имитационных моделей;
- преподавание дисциплины на основе результатов научных исследований;
- поиск и обработка информации с использованием сети Интернет;
- дистанционные образовательные технологии;
- использование на лекционных занятиях видеоматериалов и наглядных пособий;
- самостоятельная работа в системе компьютерного тестирования.

7. ПРАКТИЧЕСКАЯ ПОДГОТОВКА

Практическая подготовка в рамках учебной дисциплины реализуется при проведении лабораторных работ, предусматривающих участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.

8. ОРГАНИЗАЦИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

При обучении лиц с ограниченными возможностями здоровья и инвалидов используются подходы, способствующие созданию безбарьерной образовательной среды: технологии дифференциации и индивидуального обучения, применение соответствующих методик по работе с инвалидами, использование средств дистанционного общения, проведение дополнительных индивидуальных консультаций по изучаемым теоретическим вопросам и практическим занятиям, оказание помощи при подготовке к промежуточной аттестации.

При необходимости рабочая программа дисциплины может быть адаптирована для обеспечения образовательного процесса лицам с ограниченными возможностями здоровья, в том числе для дистанционного обучения.

Учебные и контрольно-измерительные материалы представляются в формах, доступных для изучения студентами с особыми образовательными потребностями с учетом нозологических групп инвалидов:

Для подготовки к ответу на практическом занятии, студентам с ограниченными возможностями здоровья среднее время увеличивается по сравнению со средним временем подготовки обычного студента.

Для студентов с инвалидностью или с ограниченными возможностями здоровья форма проведения текущей и промежуточной аттестации устанавливается с учетом индивидуальных психофизических особенностей (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.).

Промежуточная аттестация по дисциплине может проводиться в несколько этапов в форме рубежного контроля по завершению изучения отдельных тем дисциплины. При необходимости студенту предоставляется дополнительное время для подготовки ответа на зачете или экзамене.

Для осуществления процедур текущего контроля успеваемости и промежуточной аттестации обучающихся создаются, при необходимости, фонды оценочных средств, адаптированные для лиц с ограниченными возможностями здоровья и позволяющие оценить достижение ими запланированных в основной образовательной программе результатов обучения и уровень сформированности всех компетенций, заявленных в образовательной программе.

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Характеристика материально-технического обеспечения дисциплины соответствует требованиям ФГОС ВО.

Материально-техническое обеспечение дисциплины при обучении с использованием традиционных технологий обучения.

Наименование учебных аудиторий, лабораторий, мастерских, библиотек, спортзалов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.	Оснащенность учебных аудиторий, лабораторий, мастерских, библиотек, спортивных залов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.
119071, г. Москва, Малый Калужский переулок, дом 1, строение 3	
аудитории для проведения занятий лекционного типа	комплект учебной мебели, технические средства обучения, служащие для представления учебной информации большой аудитории: – компьютерная техника (ноутбук/компьютер); – проектор;

Наименование учебных аудиторий, лабораторий, мастерских, библиотек, спортзалов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.	Оснащенность учебных аудиторий, лабораторий, мастерских, библиотек, спортивных залов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.
аудитории для проведения практических занятий, выполнения лабораторных работ, занятий по практической подготовке, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	– экран. комплект учебной мебели, технические средства обучения, служащие для представления учебной информации большой аудитории: – компьютерная техника (ноутбук/компьютер); – проектор; – экран; – персональные компьютеры, подключенные к сети Интернет.
Помещения для самостоятельной работы обучающихся	Оснащенность помещений для самостоятельной работы обучающихся
читальный зал библиотеки:	– компьютерная техника, подключение к сети «Интернет»

Материально-техническое обеспечение учебной дисциплины при обучении с использованием электронного обучения и дистанционных образовательных технологий.

Необходимое оборудование	Параметры	Технические требования
Персональный компьютер/ноутбук/планшет, камера, микрофон, динамики, доступ в сеть Интернет	Веб-браузер	Версия программного обеспечения не ниже: Chrome 72, Opera 59, Firefox 66, Edge 79, Яндекс.Браузер 19.3
	Операционная система	Версия программного обеспечения не ниже: Windows 7, macOS 10.12 «Sierra», Linux
	Веб-камера	640x480, 15 кадров/с
	Микрофон	любой
	Динамики (колонки или наушники)	любые
	Сеть (интернет)	Постоянная скорость не менее 192 кБит/с

Технологическое обеспечение реализации программы осуществляется с использованием элементов электронной информационно-образовательной среды университета.

10. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

№ п/п	Автор(ы)	Наименование издания	Вид издания (учебник, УП, МП и др.)	Издательство	Год издания	Адрес сайта ЭБС или электронного ресурса (заполняется для изданий в электронном виде)	Количество экземпляров в библиотеке Университета
10.1 Основная литература, в том числе электронные издания							
1	Синаторов С.В.	Информационные технологии	Учебное пособие	М.: Флинта	2021	https://znanium.com/catalog/document?id=374932	-
2	Шитов В.Н.	Информатика и информационно-коммуникационные технологии в профессиональной деятельности	Учебное пособие	М: НИЦ ИНФРА-М	2022	https://znanium.com/catalog/document?id=388696	-
3	Немцова Т.И. и др.; под ред. Гагариной Л.Г.	Программирование на языке высокого уровня. Программирование на языке Object Pascal	Учебное пособие	М.: ИД ФОРУМ: НИЦ ИНФРА-М	2020	https://znanium.com/catalog/document?id=362746	-
4	Шуляк О.А.	Основы программирования	Учебно-методическая литература	М.: Флинта	2021	https://znanium.com/catalog/document?id=390158	-
5	Немцова Т.И., Голова С.Ю., Терентьев А.И.; под ред. Л.Г. Гагариной.	Программирование на языке высокого уровня. Программирование на языке С++	Учебное пособие	М.: ИД ФОРУМ: ИНФРА-М	2021	https://znanium.com/catalog/document?id=363426	-
10.2 Дополнительная литература, в том числе электронные издания							
1	Алексеев Е.Р., Чеснокова О.В.,	Free Pascal и Lazarus	Учебник	М: ДМК Пресс	2010	https://znanium.com/catalog/document?id=147218	-

	Кучер Т.В.						
2	Плотникова Н.Г.	Информатика и информационно-коммуникационные технологии (ИКТ)	Учебное пособие	М.: РИОР	2021	https://znanium.com/catalog/document?id=370445	-
3	Горбатов С.М., Тарасов Ю.С., Наумова М.Г.	Информационные технологии	Учебное пособие	М.: МИСиС	2016	https://znanium.com/catalog/document?id=371025	-
4	Федотова Е.Л.	Информационные технологии и системы	Учебное пособие	М.: Издательский Дом ФОРУМ	2022	https://znanium.com/catalog/document?id=386738	-
5	М. В. Огнева, Е. В. Кудрина	Программирование на языке С++: практический курс	Учебное пособие	М.: Издательство Юрайт	2022	https://urait.ru/bcode/492984	-
10.3 Методические материалы (указания, рекомендации по освоению дисциплины (модуля) авторов РГУ им. А. Н. Косыгина)							
1	Козлов А.М.	Технология объектно-ориентированного программирования на языке Pascal	Учебно-методическое пособие.	М.: ФГБОУ ВО «РГУ им. А.Н. Косыгина»	2020	локальная сеть университета	5
2	Козлов А.М.	Технология программирования на языке Pascal	Методические указания	М.: ФГБОУ ВО «РГУ им. А.Н. Косыгина»	2019	локальная сеть университета	5
3	Семенов А.А.	Основы объектно-ориентированного программирования в среде С++Builder	Методическое пособие	М.: ИИЦ МГУДТ	2010	локальная сеть университета	5

11. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОГО ПРОЦЕССА

11.1. Ресурсы электронной библиотеки, информационно-справочные системы и профессиональные базы данных:

№ пп	Электронные учебные издания, электронные образовательные ресурсы
1.	ЭБС «Лань» http://www.e.lanbook.com/
2.	«Znaniium.com» научно-издательского центра «Инфра-М» http://znaniium.com/
3.	Электронные издания «РГУ им. А.Н. Косыгина» на платформе ЭБС «Znaniium.com» http://znaniium.com/
4.	ЭБС «ИВИС» http://dlib.eastview.com/
5.	Образовательная платформа «ЮРАЙТ» https://urait.ru/
Профессиональные базы данных, информационные справочные системы	
1.	Scopus https://www.scopus.com (международная универсальная реферативная база данных, индексирующая более 21 тыс. наименований научно-технических, гуманитарных и медицинских журналов, материалов конференций примерно 5000 международных издательств);
2.	Научная электронная библиотека eLIBRARY.RU https://elibrary.ru (крупнейший российский информационный портал в области науки, технологии, медицины и образования);
3.	База данных в мире Academic Search Complete - обширная полнотекстовая научно-исследовательская. Содержит полные тексты тысяч рецензируемых научных журналов по химии, машиностроению, физике, биологии. http://search.ebscohost.com

11.2. Перечень программного обеспечения

№п/п	Программное обеспечение	Реквизиты подтверждающего документа/ Свободно распространяемое
1.	Windows 10 Pro, MS Office 2019	контракт № 18-ЭА-44-19 от 20.05.2019
2.	Microsoft Visual Studio	контракт № 18-ЭА-44-19 от 20.05.2019
3.	Lazarus — открытая среда разработки программного обеспечения на языке Object Pascal для компилятора Free Pascal.	Свободно распространяемое на условиях GNU General Public License.
4.	Embarcadero C++Builder RAD Studio Professional Academic Concurrent License	№ 15-02.01-2459 от 21.12.2021 Embarcadero License Certificate: #546431, #546432, #546433, #546434, #546435
5.	Code::Blocks — свободная кроссплатформенная среда разработки на C++.	Свободно распространяемое на условиях GNU General Public License v.3.
6.	Visual Studio Community	Свободно распространяемая среда разработки.
7.	Visual Studio Code	Свободно распространяемая среда разработки.

**ЛИСТ УЧЕТА ОБНОВЛЕНИЙ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ
ДИСЦИПЛИНЫ**

В рабочую программу учебной дисциплины внесены изменения/обновления и утверждены на заседании кафедры:

№ пп	год обновления РПД	характер изменений/обновлений с указанием раздела	номер протокола и дата заседания кафедры