

Подписано электронной подписью:
Вержицкий Данил Григорьевич
Должность: Директор КГПИ ФГБОУ ВО «КемГУ»
Дата и время: 2024-02-21 00:00:00

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Кузбасский гуманитарно-педагогический институт
федерального государственного бюджетного образовательного учреждения
высшего образования
«Кемеровский государственный университет»
Факультет информатики, математики и экономики

УТВЕРЖДАЮ

Декан А.В.Фомина

«10» февраля 2022 г.

Рабочая программа дисциплины

Б1.О.07 Программирование

Код, название дисциплины

Направление подготовки

09.03.03 Прикладная информатика

Код, название направления

Направленность (профиль) подготовки
Прикладная информатика в экономике

Программа бакалавриата

Квалификация выпускника

бакалавр

Форма обучения

Очная

Год набора 2021

Новокузнецк 2022

Оглавление

1	Цель дисциплины	3
1.1	Формируемые компетенции	3
1.2	Индикаторы достижения компетенций	3
1.3	Знания, умения, навыки (ЗУВ) по дисциплине	3
2	Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации	4
3.	Учебно-тематический план и содержание дисциплины	5
3.1	Учебно-тематический план	5
3.2.	Содержание занятий по видам учебной работы	7
4	Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации	13
5	Материально-техническое, программное и учебно-методическое обеспечение дисциплины	15
5.1	Учебная литература	15
5.2	Материально-техническое и программное обеспечение дисциплины	16
5.3	Современные профессиональные базы данных и информационные справочные системы	19
6	Иные сведения и (или) материалы	19
6.1.	Примерные темы письменных учебных работ	19
6.2.	Примерные вопросы и задания / задачи для промежуточной аттестации	22

1 Цель дисциплины.

В результате освоения данной дисциплины у обучающегося должны быть сформированы компетенции основной профессиональной образовательной программы бакалавриата (далее - ОПОП):

ОПК-7 – Способен разрабатывать алгоритмы и программы, пригодные для практического применения.

Содержание компетенций как планируемых результатов обучения по дисциплине см. таблицы 1 и 2.

1.1 Формируемые компетенции

Таблица 1 - Формируемые дисциплиной компетенции

Наименование вида компетенции (универсальная, общепрофессиональная, профессиональная)	Наименование категории (группы) компетенций	Код и название компетенции
общепрофессиональная		ОПК-7Способен разрабатывать алгоритмы и программы, пригодные для практического применения

1.2 Индикаторы достижения компетенций

Таблица 2 – Индикаторы достижения компетенций, формируемые дисциплиной

Код и название компетенции	Индикаторы достижения компетенции по ОПОП	Дисциплины и практики, формирующие компетенцию ОПОП
ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК 7.1 Осуществляет и обосновывает выбор стандартных алгоритмов и программных средств для реализации практических задач ОПК 7.2 Разрабатывает алгоритмы и программы для реализации прикладных практических задач	Б1.О.04 Структуры и алгоритмы компьютерной обработки данных Б1.О.07 Программирование Б2.О.01(У) Учебная практика. Ознакомительная практика Б2.О.02(П) Производственная практика. Эксплуатационная практика Б3.01(Д) Выполнение и защита выпускной квалификационной работы

1.3 Знания, умения, навыки (ЗУВ) по дисциплине

Таблица 3 – Знания, умения, навыки, формируемые дисциплиной

Код и название компетенции	Индикаторы достижения компетенции, закрепленные за дисциплиной	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК 7.1 Осуществляет и обосновывает выбор стандартных алгоритмов и программных средств для реализации практических задач ОПК 7.2 Разрабатывает	Знать: <ul style="list-style-type: none">• классификацию программных средств, языков программирования;• основные парадигмы программирования;• понятия и методы алгоритмизации;

Код и название компетенции	Индикаторы достижения компетенции, закрепленные за дисциплиной	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
	алгоритмы и программы для реализации прикладных практических задач	<ul style="list-style-type: none"> • основы и методы структурного программирования; • основные понятия объектно-ориентированного программирования; • основы теории алгоритмов и основы теории сложности; <p>Уметь:</p> <ul style="list-style-type: none"> • разрабатывать алгоритмы для решения прикладных практических задач; • разрабатывать программы для реализации прикладных практических задач; • обосновывать выбор стандартных алгоритмов для решения практических задач; • осуществлять выбор стандартных средств для программной реализации алгоритмов и программ; <p>Владеть:</p> <ul style="list-style-type: none"> • методами алгоритмизации, оценки сложности алгоритмов; • графическим способом описания алгоритмов; • методами структурного программирования; • навыками реализации алгоритмов и программ, с учетом сложности алгоритмов

2 Объём и трудоёмкость дисциплины по видам учебных занятий.

Формы промежуточной аттестации.

Таблица 4 – Объём и трудоёмкость дисциплины по видам учебных занятий

Общая трудоёмкость и виды учебной работы по дисциплине, проводимые в разных формах	Объём часов по формам обучения
	ОФО
1 Общая трудоёмкость дисциплины	468
2 Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)	
Аудиторная работа (всего):	144
в том числе:	
лекции	54
практические занятия, семинары	90

практикумы	
лабораторные работы	
в интерактивной форме	
в электронной форме	
Внеаудиторная работа (всего):	216
в том числе, индивидуальная работа обучающихся с преподавателем	213
подготовка курсовой работы/контактная работа	3
групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие групповую или индивидуальную работу обучающихся с преподавателем)	
творческая работа (эссе)	
3 Самостоятельная работа обучающихся (всего)	213
4 Промежуточная аттестация обучающегося и объём часов, выделенный на промежуточную аттестацию:	108
1 семестр – экзамен	36
2 семестр – экзамен	36
3 семестр – экзамен	36

3. Учебно-тематический план и содержание дисциплины.

3.1 Учебно-тематический план

Таблица 5 - Учебно-тематический план очной формы обучения

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость (всего час.)	Трудоёмкость занятий (час.)			Формы текущего контроля и промежуточной аттестации успеваемости
			ОФО			
			Аудиторн. занятия		СРС	
			лекц.	практ.		
Семестр 1						
	1. Основы алгоритмизации и основные понятия языка высокого уровня	41	8	8	25	
1-4	1.1 Понятия алгоритма и его свойства. Методы разработки алгоритма	13	4	4	5	Реферат
5	1.2 Классификация языков программирования. Языки высокого уровня	12	2	-	10	Реферат
5-8	1.3 Линейные программы	16	2	4	10	
	2. Основы структурного программирования	49	6	6	27	
9-14	2.1 Базовые конструкции структурного программирования	35	4	6	25	Индивидуальное задание №1
13-14	2.2 Принципы структурного программирования	4	2	-	2	

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость (всего час.)	Трудоёмкость занятий (час.)			Формы текущего контроля и промежуточной аттестации успеваемости
			ОФО			
			Аудиторн. занятия		СРС	
			лекц.	практ.		
	<i>3. Основы теории алгоритмов</i>	28	4	4	20	
15-16	3.1. Модели вычислений	14	2	2	10	Индивидуальное задание №2
17-18	3.2 Сложностные классы	14	2	2	10	Индивидуальное задание №3
19	Промежуточная аттестация - экзамен	36				Экзамен
ИТОГО по семестру 1		144	18	18	72	
Семестр 2						
	<i>4. Скалярные и нескаларные типы данные</i>	60	10	20	30	
1	4.1 Типизация. Примитивные (скалярные) типы данных языков высокого уровня	14	2	2	10	
2-6	4.2 Агрегатные (нескалярные, структурные) типы данных	24	4	10	10	Индивидуальное задание №4
7-10	4.3 Структуры данных и их реализация	22	4	8	10	Индивидуальное задание №5
	<i>5. Структурное программирование</i>	46	4	12	30	
11-15	5.1 Подпрограммы	32	2	10	20	Индивидуальное задание №6
13-16	5.2 Основы объектно-ориентированного программирования	14	2	2	10	
	<i>6. Платформа .NET</i>	38	4	4	30	
15	6.1 Архитектура и базовые принципы	12	2	-	10	
17-18	6.2 Языки платформы	28	2	4	20	Индивидуальное задание №7
	Промежуточная аттестация - экзамен	36				Экзамен
ИТОГО по семестру 2		180	18	36	90	
Семестр 3						
	<i>7. Windows-form приложения</i>	32	4	8	20	
1-2	7.1 Пользовательский интерфейс. Элементы управления.	16	2	4	10	

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость (всего час.)	Трудоёмкость занятий (час.)			Формы текущего контроля и промежуточной аттестации успеваемости
			ОФО			
			Аудиторн. занятия		СРС	
			лекц.	практ.		
3-4	7.2 Многооконные приложения	16	2	4	10	Индивидуальное задание №8
	<i>8. Языки программирования и методы трансляции</i>	73	14	28	31	
5-8	8.1 Формальные методы описания синтаксиса языка. Методы лексического анализа	22	4	8	10	
9-14	8.2 Методы синтаксического анализа языков программирования	29	6	12	11	
15-18	8.3 Семантический анализ	22	4	8	10	Индивидуальное задание №9
	Промежуточная аттестация - курсовая работа	3				Курсовая работа
	Промежуточная аттестация - экзамен	36				Экзамен
ИТОГО по семестру 3		144	18	36	51	
Всего:		468	54	90	213	

3.2. Содержание занятий по видам учебной работы

Таблица 6 – Содержание дисциплины

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
Семестр 1		
<i>Содержание лекционного курса</i>		
1	<i>Основы алгоритмизации и основные понятия языка высокого уровня</i>	
1.1	Понятия алгоритма и его свойства. Методы разработки алгоритма	<i>Понятие алгоритма. Свойства алгоритма: конечность, элементарность, дискретность, детерминированность, результативность, массовость, эффективность. Типы алгоритмов. Способы записи алгоритмов: вербальный, символьный, графический. Методы разработки: разбиение данных, разложение на подзадачи, метод последовательных приближений, рекурсия, решение обратной задачи, метод полного перебора, эвристические методы разработки, динамическое программирование.</i>
1.2	Графическое представление алгоритма.	<i>ГОСТ 19.701-90 «Схемы алгоритмов программ, данных и систем». Визуальные средства разработки.</i>

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
	Визуальное программирование	<i>Графические языки программирования: назначение, использование. Основные программные продукты.</i>
1.3	Классификация языков программирования. Языки высокого уровня	<i>Эволюция языков программирования. Классификация языков программирования. Процедурное, объектно-ориентированное, функциональное программирование. Порядок разработки и исполнения программы. Транслятор, компилятор, интерпретатор. Алфавит, синтаксис, семантика. Константы, имена, типы данных. Классификация операций. Преобразование типов данных.</i>
1.4	Линейные программы	<i>Переменные. Выражения. Структура программы. Примеры линейных алгоритмов.</i>
2	<i>Основы структурного программирования</i>	
2.1	Базовые конструкции структурного программирования	<i>Понятие структурного программирования. Следование. Ветвление. Цикл.</i>
2.2	Программирование условий и циклов	<i>Условные операторы. Операторы выбора. Виды циклов. Параметры цикла. Условия. Логические операторы. Вложенные циклы</i>
2.3	Принципы структурного программирования	<i>Принцип абстракции. Принцип формальности. Принцип модульности. Принцип иерархической упорядоченности. Нисходящее проектирование. Структурное кодирование.</i>
3	<i>Основы теории алгоритмов</i>	
3.1	Модели вычислений	<i>Понятие и основные задачи теории алгоритмов. Понятие модели вычислений. Определение алгоритма с точки зрения теории алгоритма. Конечные автоматы. Машина Тьюринга. Нормальные алгорифмы Маркова. Рекурсивные алгоритмы. Проблема разрешимости алгоритмов.</i>
3.2	Сложностные классы	<i>Временная сложность алгоритма. Трудоемкость алгоритма. Количественно-зависимые по трудоемкости алгоритмы. Параметрически-зависимые по трудоемкости алгоритмы. Количественно-параметрические по трудоемкости алгоритмы. Асимптотический анализ. Виды оценок. Классы сложности алгоритмов: P-класс, NP-класс, NP-полные задачи.</i>
<i>Содержание практических занятий</i>		
1	<i>Основы алгоритмизации и основные понятия языка высокого уровня</i>	
1.1	Алгоритмизация задач	<i>Представление алгоритмов на естественном языке</i>
1.2	Визуальные средства разработки	<i>Изучение CASE-средств разработки и графического представления алгоритмов.</i>
1.3	Построение линейных	<i>Решение простейших задач. Составление алгоритмов</i>

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
	алгоритмов	<i>линейной структуры</i>
1.4	Реализация линейных алгоритмов	<i>Программная реализация линейного алгоритма</i>
2	<i>Структурное программирование</i>	
2.1	Построение и реализация алгоритмов ветвления	<i>Разработка алгоритмов на естественном языке. Представление алгоритмов в графическом виде. Оптимизация памяти и количества операций. Реализация алгоритмов.</i>
2.2	Построение циклических алгоритмов. Циклы с пост- и пред-условием	<i>Решение задач для алгоритмов с циклической структурой</i>
2.3	Реализация циклических алгоритмов. Циклы со счетчиком	<i>Решение задач для алгоритмов с циклической структурой</i>
3	<i>Основы теории алгоритмов</i>	
3.1	Построение машины Тьюринга и нормальных алгорифмов	<i>Построение простейших машин Тьюринга. Построение нормальных алгорифмов Маркова</i>
3.2	Оценка временной сложности алгоритмов	<i>Оценка временной сложности, с использованием асимптотического анализа. Оценка временной сложности, исходя из количества операций</i>
Промежуточная аттестация - экзамен		
Семестр 2		
<i>Содержание лекционного курса</i>		
4	<i>Структурные типы данных</i>	
4.1	Типизация. Примитивные (скалярные) типы данных языков высокого уровня	<i>Понятие типизации. Классификация типизаций. Статическая и динамическая типизация. Явная и неявная типизация. Строгая и нестрогая типизация. Примитивные типы данных</i>
4.2	Составные типы данных. Указатели. Ссылки	<i>Массивы. Статические массивы. Объявление и инициализация массивов. Динамические массивы. Динамическое выделение памяти. Объявление динамического массива. Освобождение памяти. Переопределение размера динамического массива. Вектора. Строковые типы данных. Функции со строковыми переменными. Указатели. Ссылки. Объявление указателей. Инициализация указателей. Разыменовывание указателя. Указатели на указатели. Задание массива через указатели. Указатели на функции.</i>
4.2	Файловый тип данных	<i>Виды файловых переменных. Виды доступа к файлу. Действия с файлами. Последовательность работы с файлом.</i>
4.3	Структуры данных	<i>Классификация структур данных. Основные структуры данных. Стеки. Очереди. Связанные списки. Графы. Деревья. Хэш-таблицы</i>

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
4.3	Реализация структур данных	<i>Действия со структурами данных. Реализация структур данных.</i>
5	<i>Структурное программирование</i>	
5.1	Подпрограммы	<i>Подпрограммы. Глобальные и локальные переменные. Формальные и фактические параметры.</i>
5.2	Основы объектно-ориентированного программирования	<i>Объектно-ориентированная парадигма. Абстракция. Наследование. Инкапсуляция. Полиморфизм. Класс. Объект. Особенности реализации. Объектно-ориентированные языки</i>
6	<i>Платформа .NET</i>	
6.1	Архитектура и базовые принципы	<i>Концепция. Архитектура Поддержка нескольких языков. Кроссплатформенность. Библиотеки классов. Разнообразие технологий. .NETFrameworkу .NETCore. Управляемые и неуправляемый код.</i>
6.2	Языки платформы	<i>Среды разработки. Языки программирования в .NET. Язык программирования С#. Применение. Особенности.</i>
<i>Содержание практических занятий</i>		
4	<i>Скалярные и не скалярные типы данные</i>	
4.1	Разработка программного кода с явным и неявным преобразованием типов	<i>Решение практических задач и программная реализация задач, требующих преобразование типов данных с учетом вида типизации языка программирования</i>
4.2	Реализация статических массивов	<i>Решение практических задач на хранение, заполнение и обработку статических одномерных и многомерных массивов. Реализация решения математических задач с матрицами.</i>
4.2	Реализация динамических массивов	<i>Решение задач с динамическим выделением памяти для хранения. Реализация динамических массивов и векторов</i>
4.2	Строковые переменные	<i>Реализация решения задач с обработкой строк</i>
4.2	Указатели и ссылки	<i>Программная реализация объявления и разыменовывания указателей. Создание и обработка указателей на массив и указателей на указатели.</i>
4.2	Файловые переменные	<i>Реализация записи и чтения из файла</i>
4.3	Реализация стека и очереди	<i>Программная реализация структуры данных и операций с ней с использованием указателей и записей</i>
4.3	Программная реализация бинарного дерева	<i>Реализация структуры дерева</i>
4.3	Алгоритмы решения практических задач, основанные на использовании деревьев	<i>Решение алгоритма сортировки и кодирования информации</i>
4.3	Реализация вычисление хэш-таблиц	<i>Реализация алгоритма хэширования</i>
5	<i>Структурное программирование</i>	
5.1	Реализация подпрограмм	<i>Использование принципов структурного программирования для выделения подзадач. Реализация решения подзадач в виде</i>

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
		<i>подпрограмм</i>
5.1	Передача в подпрограмму параметров через указатели	<i>Реализация подпрограмм, в которых параметрами являются указатели</i>
5.1	Передача в подпрограмму массива	<i>Реализация подпрограмм, в которых параметрами являются массивы</i>
5.1	Применение принципов структурного программирования для решения практических задач	<i>Выявление в алгоритме решения подзадач, структур. Определение синтаксических лексем языка для реализации</i>
5.1	Реализация рекурсивных подпрограмм	<i>Реализация решения задач, с использованием рекурсии</i>
5.2	Применение принципов объектно-ориентированного программирования	<i>Разработка классов, атрибутов и методов. Разработка наследуемых классов</i>
6	<i>Платформа .NET</i>	
6.2	Синтаксис языка C#	<i>Решение задач структурного программирования на языке C#</i>
6.2	Использование структурных данных на языке C#	<i>Программная реализация задач, требующих применение массивов, строковых данных и других структурных типов переменных.</i>
	Промежуточная аттестация - экзамен	
	Семестр 3	
	<i>Содержание лекционного курса</i>	
7	<i>Windows-form приложения</i>	
7.1	Пользовательский интерфейс. Элементы управления.	Создание графического приложения. Объекты и классы формы. Основные управляющие объекты. Свойства объектов. События элементов формы. Чтение и запись в элементы формы.
7.2	Многооконные приложения	MDI приложение. Создание родительских MDI-форм. Создание дочерних MDI-форм. Определение активных дочерних форм. Отправка данных
8	<i>Языки программирования и методы трансляции</i>	
8.1	Основные термины и определения	Фазы трансляции и выполнения программ. Модульное программирование. Библиотека объектных модулей. Структура транслятора. Связывание. Сравнительная характеристика языков программирования.
8.1	Лексический анализ	Сущность лексического анализа. Лексемы. Комментарии. Идентификаторы. Константы. Служебные слова. Знаки операций. Простейший лексический анализатор. Конечные автоматы в лексическом анализе.
8.2	Синтаксический анализ.	Роль и место формальных грамматик в синтаксическом

№ п/п	Наименование раздела, темы дисциплины	Содержание занятия
	Формальные грамматики и языки	анализе. Взаимосвязь синтаксиса и формальных грамматик. Представление синтаксиса в формальных грамматиках. Альтернатива. Повторение. Вложенность. Приоритеты. Необязательные элементы. Разделители. Ограничители. Отношения между символами формальной грамматики
8.2	Нисходящий разбор	Основные идеи нисходящего разбора. Нисходящий разбор с возвратами. Нисходящий разбор и стековые автоматы. Рекурсивный спуск.
8.2	Восходящий разбор	Восходящий разбор. Свертка-перенос. Простое предшествование
8.3	Семантический анализ	Особенности семантики и семантического анализа. Семантика программы. Синтез выходного кода. Генерация кода. Интерпретация. Границы лексики, синтаксиса и семантики
8.3	Внутреннее представление программы	Внутреннее представление Си-программы Внутреннее представление Java-программы
<i>Содержание практических занятий</i>		
7	<i>Windows-form приложения</i>	
7.1	Реализация изменения свойств объектов Windows-form	Объекты Windows-form. Свойства объектов. Программное изменение свойств
7.1	Реализация пользовательских интерфейсов в Windows приложениях	Решение практических задач
7.2	Реализация многооконных приложений	Создание объектов и связь между формами
7.2	Передача данных между формами	Создание многооконных приложений
8	<i>Языки программирования и методы трансляции</i>	
8.1	Проведение лексического анализа языков программирования	Лексический анализ языка программирования. Построение конечных автоматов. Выявление лексических объектов.
8.2	Изучение формальных грамматик языков	Представление синтаксиса в формальных грамматиках.
8.2	Проведение нисходящего разбора	Нисходящий анализ языка программирования
8.2	Проведение восходящего разбора	Восходящий анализ языка программирования
8.3	Проведение семантического анализа	Внутреннее представление Си-программы
<i>Промежуточная аттестация – курсовая работа</i>		
<i>Промежуточная аттестация - экзамен</i>		

4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации.

Для положительной оценки по результатам освоения дисциплины обучающемуся необходимо выполнить все установленные виды учебной работы. Оценка результатов работы обучающегося в баллах (по видам) приведена в таблице 7.

Пример заполнения таблицы

Таблица 7.1 - Балльно-рейтинговая оценка результатов учебной работы обучающихся по видам(БРС) 1 семестр

Учебная работа (виды)	Сумма баллов	Виды и результаты учебной работы	Оценка в аттестации	Баллы (17 недель)
Текущая учебная работа в семестре (Посещение занятий по расписанию и выполнение заданий)	60	Лекционные занятия (конспект) (9 занятий)	0,5 балла посещение 1 лекционного занятия	4
		Лабораторные работы (отчет о выполнении лабораторной работы) (9 работ).	2 балл - посещение 1 практического занятия и выполнение работы на 51-65% 3 балла – посещение 1 занятия и существенный вклад на занятии в работу всей группы, самостоятельность и выполнение работы на 85,1-100%	18 - 27
		Индивидуальные задания(отчет о выполнении) (3 работы)	За одну ИД от 2 до: 3 балла (выполнено 66 - 85% заданий) 5 баллов (выполнено 86 - 100% заданий)	6 - 25
		Реферат (по разделу 1 или 2 на выбор)	3 балла (пороговое значение) 4 баллов (максимальное значение)	3 - 4
Итого по текущей работе в семестре				31 - 60
Промежуточная аттестация (экзамен)	40 (100% /баллов приведенной шкалы)	Теоретический вопрос 1.	3 балла (пороговое значение) 10 баллов (максимальное значение)	5 - 10
		Теоретический вопрос 2.	3 балла (пороговое значение) 5 баллов (максимальное значение)	7 - 14
		Решение задачи 1.	3 балла (пороговое значение) 5 баллов (максимальное значение)	8 - 16
Итого по промежуточной аттестации (экзамену)				(51 – 100% по приведенной шкале) 20 – 40 б.
Суммарная оценка по дисциплине: Сумма баллов текущей и промежуточной аттестации				51 – 100 б.

Таблица 7.2 - Балльно-рейтинговая оценка результатов учебной работы обучающихся по видам(БРС) 2 семестр

Учебная работа (виды)	Сумма баллов	Виды и результаты учебной работы	Оценка в аттестации	Баллы (17 недель)
Текущая учебная работа в семестре (Посещение занятий по расписанию и выполнение заданий)	60	Лекционные занятия (конспект) (9 занятий)	0,5 балла посещение 1 лекционного занятия	4
		Лабораторные работы (отчет о выполнении лабораторной работы) (18 работ).	1 балл - посещение 1 практического занятия и выполнение работы на 51-65% 2 балла – посещение 1 занятия и существенный вклад на занятии в работу всей группы, самостоятельность и	18 - 36

			выполнение работы на 85,1-100%	
		Индивидуальные задания (отчет о выполнении) (4 работы)	За одну КР от 2 до: 3 баллов (выполнено 51 - 65% заданий) 4 балла (выполнено 66 - 85% заданий) 5 баллов (выполнено 86 - 100% заданий)	8 - 20
Итого по текущей работе в семестре				31 - 60
Промежуточная аттестация (экзамен)	40 (100% /баллов приведен ной шкалы)	Теоретический вопрос 1.	3 балла (пороговое значение) 10 баллов (максимальное значение)	5 - 10
		Теоретический вопрос 2.	3 балла (пороговое значение) 5 баллов (максимальное значение)	7 - 14
		Решение задачи 1.	3 балла (пороговое значение) 5 баллов (максимальное значение)	8 - 16
Итого по промежуточной аттестации (экзамену)				(51 – 100% по приведенной шкале) 20 – 40 б.
Суммарная оценка по дисциплине: Сумма баллов текущей и промежуточной аттестации				51 – 100 б.

Таблица 7.3 - Балльно-рейтинговая оценка результатов учебной работы обучающихся по видам(БРС) 3 семестр

Учебная работа (виды)	Сумма баллов	Виды и результаты учебной работы	Оценка в аттестации	Баллы (17 недель)
Текущая учебная работа в семестре (Посещение занятий по расписанию и выполнение заданий)	60	Лекционные занятия (конспект) (9 занятий)	0,5 балла посещение 1 лекционного занятия	4
		Лабораторные работы (отчет о выполнении лабораторной работы) (18 работ).	1 балл - посещение 1 практического занятия и выполнение работы на 51-65% 2 балла – посещение 1 занятия и существенный вклад на занятии в работу всей группы, самостоятельность и выполнение работы на 85,1-100%	18 - 36
		Индивидуальные задания (отчет о выполнении) (2 работы)	За одну КР от 2 до: 4 балла (выполнено 66 - 85% заданий) 5 баллов (выполнено 86 - 100% заданий)	2 - 10
		Курсовая работа	5 баллов (пороговое значение) 10 баллов (максимальное значение)	5 - 10
Итого по текущей работе в семестре				31 - 60
Промежуточная аттестация (экзамен)	40 (100% /баллов приведен ной шкалы)	Теоретический вопрос 1.	3 балла (пороговое значение) 10 баллов (максимальное значение)	5 - 10
		Теоретический вопрос 2.	3 балла (пороговое значение) 5 баллов (максимальное значение)	7 - 14
		Решение задачи 1.	3 балла (пороговое значение) 5 баллов (максимальное значение)	8 - 16
Итого по промежуточной аттестации (экзамену)				(51 – 100% по приведенной шкале) 20 – 40 б.
Суммарная оценка по дисциплине: Сумма баллов текущей и промежуточной аттестации				51 – 100 б.

Таблица 7.4 - Балльно-рейтинговая оценка результатов учебной работы обучающихся по курсовой работе (БРС)

Учебная работа (виды)	Сумма баллов	Виды и результаты учебной работы	Оценка в аттестации	Баллы (17 недель)
Курсовая работа	80	Работа над курсовой (посещение консультаций, ход работы)	28 балла (пороговое значение) 50 баллов (максимальное значение)	28 - 50
		Результаты программирования и алгоритмизации	10 балла (пороговое значение) 20 баллов (максимальное значение)	10 - 20
		Оформление работы	3 балла (пороговое значение) 10 баллов (максимальное значение)	3 - 10
Итого по текущей работе в семестре				41 - 80
Промежуточная аттестация (курсовая работа)	20 (100% /баллов приведенной шкалы)	Защита курсовой работы	3 балла (пороговое значение) 10 баллов (максимальное значение)	10 - 20
Итого по промежуточной аттестации (курсовой работе)				(51 – 100% по приведенной шкале) 10 – 20 б.
Суммарная оценка по дисциплине: Сумма баллов текущей и промежуточной аттестации				51 – 100 б.

5 Материально-техническое, программное и учебно-методическое обеспечение дисциплины.

5.1 Учебная литература

Основная учебная литература

Трофимов, В. В. Алгоритмизация и программирование : учебник для академического бакалавриата / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — Москва : Издательство Юрайт, 2019. — 137 с. — (Бакалавр. Академический курс. Модуль). — ISBN 978-5-534-07834-3. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://biblio-online.ru/bcode/423824> (дата обращения: 26.11.2019)

Дополнительная учебная литература

Крупский, В. Н. Теория алгоритмов. Введение в сложность вычислений : учебное пособие для бакалавриата и магистратуры / В. Н. Крупский. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 117 с. — (Авторский учебник). — ISBN 978-5-534-04817-9. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://biblio-online.ru/bcode/444131> (дата обращения: 26.11.2019).

Судоплатов, С. В. Математическая логика и теория алгоритмов : учебник и практикум для академического бакалавриата / С. В. Судоплатов, Е. В. Овчинникова. — 5-е изд., стер. — Москва : Издательство Юрайт, 2019. — 255 с. — (Высшее образование). —

5.2 Материально-техническое и программное обеспечение дисциплины.

Учебные занятия по дисциплине проводятся в учебных аудиториях НФИ КемГУ

Таблица 8 – Информационные технологии и программное обеспечение аудиторных занятий и самостоятельной работы

Наименование помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом, в том числе помещения для самостоятельной работы	Перечень основного оборудования, учебно-наглядных пособий и используемого программного обеспечения	Адрес (местоположение) помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом
229 а Учебная аудитория для проведения: - занятий лекционного типа;	Специализированная (учебная) мебель: доска меловая, парты. Оборудование: микшер-усилитель, системный блок, экран настенный с электроприводом, проектор, акустическая система. Используемое программное обеспечение: MSWindows (Microsoft Imagine Premium 3 year по лицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО). Интернет с доступом в ЭИОС.	654079, Кемеровская область, г. Новокузнецк, пр-кт Metallургов, д. 19
615 Учебная аудитория (мультимедийная) для проведения: - занятий лекционного типа;	Специализированная (учебная) мебель: доска меловая, кафедра, столы, стулья. Оборудование для презентации учебного материала: стационарное - компьютер, экран, проектор, акустическая система (колонки). Используемое программное обеспечение: UbuntuLinux (свободно распространяемое ПО), LibreOffice (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО). Интернет с обеспечением доступа в ЭИОС.	654079, Кемеровская область, г. Новокузнецк, пр-кт Metallургов, д. 19

<p>509 Компьютерный класс. Учебная аудитория (мультимедийная) для проведения:</p> <ul style="list-style-type: none"> - занятий семинарского (практического) типа; - групповых и индивидуальных консультаций; - самостоятельной работы; - текущего контроля и промежуточной аттестации; 	<p>Специализированная (учебная) мебель: доска меловая, кафедра, столы, стулья, Оборудование для презентации учебного материала: стационарное - компьютер преподавателя, экран, проектор.</p> <p>Оборудование: стационарное-компьютеры для обучающихся (18 шт.).</p> <p>Используемое программное обеспечение: MSWindows (MicrosoftImaginePremium 3 year по лицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО), BloodshedDevC++ 4.9.9.2 (свободно распространяемое ПО), FoxitReader (свободно распространяемое ПО), Firefox 14 (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО), Java (бесплатная версия), Opera 12 (свободно распространяемое ПО), MicrosoftVisualStudio (MicrosoftImaginePremium 3 year по лицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.).</p> <p>Интернет с обеспечением доступа в ЭИОС.</p>	<p>654079, Кемеровская область, г. Новокузнецк, пр-кт Metallургов, д. 19</p>
<p>502 Компьютерный класс. Учебная аудитория (мультимедийная) для проведения:</p> <ul style="list-style-type: none"> - занятий семинарского (практического) типа; - групповых и индивидуальных консультаций; - самостоятельной работы; - текущего контроля и промежуточной аттестации; 	<p>Специализированная (учебная) мебель: доска меловая, столы компьютерные, стулья.</p> <p>Оборудование для презентации учебного материала: стационарное - компьютер, экран, проектор.</p> <p>Оборудование: стационарное – компьютеры для обучающихся (16 шт.).</p> <p>Используемое программное обеспечение: MSWindows (MicrosoftImaginePremium 3 year по лицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО), FoxitReader (свободно распространяемое ПО), Firefox 14 (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО), Java (бесплатная версия), MicrosoftVisualStudio (MicrosoftImaginePremium 3 year по</p>	<p>654079, Кемеровская область, г. Новокузнецк, пр-кт Metallургов, д. 19</p>

	сублицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.) Интернет с обеспечением доступа в ЭИОС.	
501 Компьютерный класс. Учебная аудитория (мультимедийная) для проведения: - занятий лекционного типа; - занятий семинарского (практического) типа; - курсового проектирования (выполнения курсовых работ); - групповых и индивидуальных консультаций; - текущего контроля и промежуточной аттестации;	Специализированная (учебная) мебель: доска меловая, кафедра, столы компьютерные, стулья. Оборудование для презентации учебного материала: стационарное - компьютер преподавателя, экран, проектор. Оборудование: стационарное - компьютеры для обучающихся (17 шт.). Используемое программное обеспечение: MSWindows (Microsoft Imagine Premium 3 year по сублицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО), FoxitReader (свободно распространяемое ПО), Firefox 14 (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО), Java (бесплатная версия), Eclipse (свободно распространяемое ПО), Microsoft Visual Studio (Microsoft Imagine Premium 3 year по сублицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.) Интернет с обеспечением доступа в ЭИОС	654079, Кемеровская область, г. Новокузнецк, пр-кт Metallургов, д. 19
508 Компьютерный класс. Учебная аудитория (мультимедийная) для проведения: - групповых и индивидуальных консультаций; - текущего контроля и промежуточной аттестации.	Специализированная (учебная) мебель: доска меловая, кафедра, столы, стулья. Оборудование для презентации учебного материала: стационарное - компьютер преподавателя, проектор, экран. Оборудование: стационарное - компьютеры для обучающихся (18 шт.). Используемое программное обеспечение: MSWindows (Microsoft Imagine Premium 3 year по сублицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО), Bloodshed DevC++ 4.9.9.2 (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО),	654079, Кемеровская область, г. Новокузнецк, пр-кт Metallургов, д. 19

	Java (бесплатная версия), Microsoft Visual Studio (Microsoft Imagine Premium 3 year по лицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.). Интернет с обеспечением доступа в ЭИОС.	
--	---	--

5.3 Современные профессиональные базы данных и информационные справочные системы.

Перечень СПБД и ИСС по дисциплине

1. CITForum.ru - on-line библиотека свободно доступных материалов по информационным технологиям на русском языке - <http://citforum.ru>
2. Научная электронная библиотека eLIBRARY.RU – крупнейший российский информационный портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты - www.elibrary.ru
3. Единое окно доступа к образовательным ресурсам - <http://window.edu.ru/>

6 Иные сведения и (или) материалы.

6.1. Примерные темы письменных учебных работ

Темы реферата

Тема 1.1 Понятия алгоритма и его свойства. Методы разработки алгоритма

1. Достоинства и недостатки метода разработки алгоритмов «Решение обратных задач»
2. Достоинства и недостатки эвристического метода разработки алгоритмов
3. Особенности динамического программирования. Применяемые технологии
4. Современное состояние технологий визуального программирования
5. Применение технологий визуального программирования при обучении программированию
6. Современные визуальные средства разработки алгоритмов

Тема 1.2 Классификация языков программирования. Языки высокого уровня

1. Современное применение языков низкого уровня
2. Языки логического программирования
3. Интерпретаторные языки программирования
4. Скриптовые языки программирования
5. Функциональное программирование
6. Языки символьной обработки

Темы индивидуальных заданий

Индивидуальное задание №1

1. Определить для каждой задачи вид алгоритма (циклический, линейные, ветвление). Для циклических алгоритмов определить наиболее подходящий тип цикла. Обосновать разницу между решениями с использованием различных типов цикла.

2. Решите задачи, реализуемые циклическими алгоритмами, представив программный код, графическое представление алгоритма и обоснование выбора типа данных. Используйте, если возможно, два варианта решения, с различными типами цикла.

Задачи:

1. Преобразовать введенное число в другое число, цифры которого будут следовать в обратном порядке по сравнению с введенным числом.
2. Сгенерировать случайное трехзначное число, оканчивающееся на ноль.
3. Требуется определить факториал числа, которое ввел пользователь
4. Найти сумму и произведение цифр числа.
5. Обменять значения двух переменных, используя третью (буферную) переменную.
6. Определить из каких цифр состоит число
7. Требуется написать программу, определяющую по координатам точки, в какой четверти она находится.
8. Вывести на экран ряд чисел Фибоначчи, состоящий из n элементов.
9. Определить какое из трех, введенных пользователем, чисел максимальное и вывести его на экран.
10. Для каждого натурального числа в промежутке от m до n вывести все делители, кроме единицы и самого числа. m и n вводятся с клавиатуры.
11. Написать программу, которая находит все комбинации из трех чисел до определенного предела, которые в сумме дают другое число.
12. Вводится четырехзначное число $(abcd)$. Вывести сумму $ab + cd$.

Индивидуальное задание №2

1. Построить машину Тьюринга стирающую 1.
2. Построить машину Тьюринга, инвертирующую бинарное число.
3. Построить машину Тьюринга, стирающую подряд идущие одинаковые символы
4. Построить машину Тьюринга, которая вычисляет модуль разности любых двух натуральных чисел.
5. Построить машину Тьюринга, которая вычисляет остаток от деления заданного конструктивного натурального числа на 5.
6. Дано: конечная последовательность 0 и 1 (001101011101). Необходимо выписать их после данной последовательности, через пустую ячейку, а в данной последовательности заменить их на 0. Например: Из 001101011101 получим 0000000000001111111.

Индивидуальное задание №3

Оценить временную сложность алгоритма. Запрограммировать алгоритм.

Варианты:

1. поиск минимального значения введенной последовательности;
2. поиск максимального значения введенной последовательности;
3. транспонирование матрицы;
4. произведение векторов;
5. сумма элементов введенной последовательности;
6. произведение матрицы на строку;
7. поиск произвольного минора матрицы;

8. вычисление геометрической последовательности и суммы n членов;
9. скалярное произведение векторов;
10. нормировка вектора.

Индивидуальное задание №4

1. В файлах A и B содержатся матрицы чисел произвольного размера, записанные как одномерный массив. Известно, что существует произведение AB и BA. Требуется написать код определяющий размерность матриц.

2. В файле A содержится стихотворение. В файле B зашифрованный текст с использованием книжного шифра (каждая буква зашифрована строка-символ в строке). Требуется написать код дешифрования и записи результата в файл C.

3. В файле A содержится матрица чисел произвольного размера, записанная как одномерный массив. В файлах B, C, D содержатся матрицы чисел, представленные в виде двумерного массива. Одна из этих матриц является обратной матрице A. Требуется написать код, определяющий обратную матрицу.

4. В файле A содержится список студентов. В файле B список предметов. В файл C внесены оценки обучающихся, причем строка соответствует студенту, а столбец предмету. Оценки отделены пробелами. Если оценка отсутствует, то ничего не ставится и между двумя оценками оказывается два пробела. Требуется написать код, определяющий средний балл обучающегося, средний балл по предмету и список долгов каждого обучающегося.

Индивидуальное задание №5

Реализовать программу, выводящую на экран элементы бинарного дерева, проводящую сортировку элементов бинарного дерева (сортировку массива), удаление листьев или корней бинарного дерева.

Индивидуальное задание №6

Реализовать программу для работы с введенными массивами произвольной длины. При этом все действия с массивами должны быть реализованы через подпрограммы. Обосновать реализованное структурирование кода. Представить блок-схемы основной программы и подпрограмм.

Индивидуальное задание №7

1. Реализуйте вычисление корней квадратного уравнения на F#
2. Реализуйте функцию вычисления кубов последовательности на F#
3. Реализуйте вывод геометрической последовательности на F#
4. Реализуйте вычисление факториала, с использованием цикла и рекурсивной функции на F#

Индивидуальное задание №8

Реализовать программное приложение вычисления функции (по вариантам) в указанных пределах и её отображение в виде графика. Приложение должно поддерживать интерфейс в виде windows-form, причем график должен выводиться на отдельном окне. На окне ввода исходных данных должно быть предложен выбор параметров вывода графика (наличие сетки, подписей по осям, шаг по осям, отображение максимума и

минимума), вычисления максимальных, минимальных и средних значений и отображения данных на экране.

Индивидуальное задание №9

Проведение синтаксического и семантического анализа языка программирования по выбору: C++, C#, Java, Python, F#, Delphi, VisualBasic.

Темы курсовой работы

Примерные темы курсовой работы:

1. Реализация программного приложения для вычисления основных статистических зависимостей и показателей данных
2. Реализация программного приложения для составления зашифрованных сообщений методом замены
3. Реализация программного приложения для вычисления корней системы линейных алгебраических уравнений методом Гаусса
4. Реализация программного приложения для вычисления максимума нелинейной функции методом золотого сечения
5. Реализация программного приложения для построения геометрических плоских фигур и вычисления их площади и периметра
6. Реализация программного приложения для вычисления объемов, площади поверхности, площадей граней трехмерных фигур
7. Реализация программного приложения для статистического анализа тестов
8. Реализация программного приложения для составления словаря иностранных слов
9. Реализация программного приложения для вычисления сложности пароля и хранения паролей
10. Реализация программного приложения для игры в крестики-нолики

Выбор темы курсовой работы производится в начале семестра. Тема курсовой работы выбирается из предложенного списка или предлагается обучающимся (при удовлетворении требований к курсовой работе).

6.2. Примерные вопросы и задания / задачи для промежуточной аттестации

Семестр 1

Таблица 9.1 - Примерные теоретические вопросы и практические задания к экзамену

Разделы и темы	Примерные теоретические вопросы	Примерные практические задания
1. Основы алгоритмизации и основные понятия языка высокого уровня		
1.1 Понятия алгоритма и его свойства. Методы разработки алгоритма	1. Виды представления алгоритмов 2. Понятие алгоритма. Свойства алгоритмов 3. Визуальное программирование.	Построить блок-схему линейного алгоритма вычисления арифметической прогрессии

	Графические языки программирования 4. Визуальное программирование. Визуальные средства разработки	
1.2 Классификация языков программирования. Языки высокого уровня	5. Классификации языков программирования 6. Виды операций и операторов. Последовательность выполнения 7. Явное преобразование типов 8. Неявное преобразование типов Компилятор, транслятор 9. Интерпритаторные языки	
1.3 Линейные программы	10. Понятие переменной. Типы обращения 11. Линейные операторы	С клавиатуры введены вещественные числа. Вывести числа, округленные до 2 знака после запятой (при этом использовать округление вверх, округление вниз и округление по правилам, по выбору пользователя) Найти уравнение прямой по двум точкам
2. Основы структурного программирования		
2.1 Базовые конструкции структурного программирования	12. Виды циклов 13. Логический тип данных. Операции с логическими типами данных 14. Ветвление 15. Вложенные циклы	Реализовать решение системы алгебраических уравнений (размерность – 2). С клавиатуры вводятся коэффициенты. Вывести таблицу сложения, таблицу умножения и таблицы стандартных логических функций в 2-ичной системе исчисления. Вывести таблицу умножения в 6-ричной системе исчисления
2.2 Средства организации модульности в языках высокого уровня	16. Принцип модульности 17. Принцип иерархической упорядоченности	
3. Основы теории алгоритмов		
3.1. Модели вычислений	18. Теория алгоритмов. Цели и задачи 19. Примитивно-рекурсивная функция, частично-рекурсивная функция 20. Конечный автомат 21. Машина Тьюринга 22. Нормальные алгорифмы Маркова	Реализовать нормальный алгорифм Маркова для инвертирования бинарного числа Построить машину Тьюринга для сложения двух чисел представленных на ленте
3.2 Сложностные классы	23. Классы сложности алгоритмов 24. Временная сложность	Вычислить сложность умножения двух матриц

	алгоритма 25.Емкостная сложность алгоритмов 26.Количественно-зависимые по трудоемкости алгоритмы 27.Параметрически-зависимые по трудоемкости алгоритмы 28.Количественно-параметрические по трудоемкости алгоритмы 29.Асимптотический анализ функций 30.NP – полные задачи	
--	---	--

Семестр 2

Таблица 9.2 - Примерные теоретические вопросы и практические задания к экзамену

Разделы и темы	Примерные теоретические вопросы	Примерные практические задания
4. Скалярные и нескаларные типы данные		
4.1 Типизация. Примитивные (скалярные) типы данных языков высокого уровня	1. Статическая типизация 2. Явная и неявная типизация	
4.2 Агрегатные (нескалярные, структурные) типы данных	3. Статические массивы 4. Динамические массивы 5. Указатели 6. Ссылки 7. Указатели на указатели 8. Строковые типы данных 9. Варианты реализации строк	<p>Реализовать свертку трехмерного массива (5x5x5) в одномерный. Значения ввести с клавиатуры.</p> <p>Реализовать алгоритм вычисления средних оценок списка обучающихся, где количество обучающихся задается пользователем в начале работы, а количество оценок у каждого обучающегося отличается.</p> <p>Дан указатель: <code>double **p = 0;</code> Реализуйте заполнение и вывод данных.</p>
4.3 Структуры данных и их реализация	10. Стек и очередь 11.Граф	<p>Используя массивы, указатели и структуры, реализуйте динамическую структуру однонаправленного связанного списка и действия с ней:добавление элемента в начало, добавление элемента в конец, добавление элемента в середину списка, ввод списка, удаление первого элемента, удаление последнего элемента, удаление элемента из середины списка, вычисление длины списка</p> <p>Используя массивы, указатели и структуры, реализуйте вывод и заполнение динамической структуры очередь</p>

5. Структурное программирование		
5.1 Подпрограммы	12. Формальные и фактические параметры 13. Подпрограмма	Написать подпрограмму, реализующую поиск гласных в слове
5.2 Основы объектно-ориентированного программирования	14. Объектно-ориентированная парадигма 15. Объектно-ориентированные языки	
6. Платформа .NET		
6.1 Архитектура и базовые принципы	16. Платформа .NET. Концепция 17. Платформа .NET. Кроссплатформенность	
6.2 Языки платформы	18. Среды разработки Платформа .NET 19. Семейство языков C	Реализовать программу вычисления корней функции на языке платформы .NET

Семестр 3

Таблица 9.3 - Примерные теоретические вопросы и практические задания к экзамену

Разделы и темы	Примерные теоретические вопросы	Примерные практические задания
7. Windows-form приложения		
7.1 Пользовательский интерфейс. Элементы управления.	1. Управляющие объекты графически приложений 2. События элементов формы	Реализовать графическое приложение визуализации графиков тригонометрических функций Реализовать графическое приложение визуализации неявных функций двух переменных
7.2 Многооконные приложения	3. MDI-приложение 4. Активные дочерние формы	Реализовать приложение с двумя формами для ввода данных и выбора места сохранения
8. Языки программирования и методы трансляции		
8.1 Формальные методы описания синтаксиса языка. Методы лексического анализа	5. Фазы трансляции 6. Лексический анализатор 7. Лексемы 8. Конечные автоматы в лексическом анализе	Выявить в коде программы лексические объекты
8.2 Методы синтаксического анализа языков программирования	9. Формальные грамматики 10. Необязательные элементы 11. Нисходящий разбор 12. Восходящий разбор	Выделить необязательные элементы синтаксиса в языке
8.3 Семантический анализ	13. Семантика программы 14. Синтез выходного языка	Определить семантику языка

Составитель (и): Штейнбрехер О.А., канд. техн. наук, доцент кафедры информатики и вычислительной техники им. В.К. Буторина