

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего
образования «Кабардино-Балкарский государственный университет
им. Х.М. Бербекова» (КБГУ)

ИНСТИТУТ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА И ЦИФРОВЫХ ТЕХНОЛОГИЙ
КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

СОГЛАСОВАНО

Руководитель образовательной
программы  М.М. Лафишева

« 12 »  2023г.

УТВЕРЖДАЮ

Директор института
 А.Х. Шапигов

« 12 »  2023г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«ЯЗЫКИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ»

02.03.02 Фундаментальная информатика и информационные технологии
(код и наименование направления подготовки)

«Проектирование систем искусственного интеллекта»
(наименование профиля подготовки)

Бакалавр

Квалификация (степень) выпускника

Очная

Форма обучения

Нальчик - 2023

Рабочая программа дисциплины «Языки и методы программирования» / сост. А.Л. Нагоров – Нальчик: КБГУ, 2023. – 63 с.

Рабочая программа дисциплины предназначена для студентов очной формы обучения направления подготовки 02.03.02 Фундаментальная информатика и информационные технологии, профиль «Проектирование систем искусственного интеллекта» в 1,2 семестрах 1 курса.

Рабочая программа составлена с учетом Федерального государственного образовательного стандарта высшего образования по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии, утверждённого приказом Министерства образования и науки Российской Федерации от 23 августа 2017 г. N 808 (с изменениями и дополнениями от: 26 ноября 2020 г., 8 февраля 2021 г., 19 июля 2022 г.) (Зарегистрировано в министерстве юстиции Российской Федерации 14 сентября 2017 г. Регистрационный N 48190).

Содержание

1. Цель и задачи освоения дисциплины.....	4
2. Место дисциплины в структуре ОПОП ВО	4
3. Требования к результатам освоения дисциплины (модуля).....	5
4. Содержание и структура дисциплины.....	6
5. Оценочные материалы для текущего и рубежного контроля успеваемости и промежуточной аттестации	13
6. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности.....	44
7. Учебно-методическое обеспечение дисциплины (модуля)	46
7.1. Нормативно-законодательные акты	46
7.2. Основная литература	47
7.3. Дополнительная литература.....	47
7.4. Периодические издания	48
7.5. Интернет-ресурсы	48
7.6. Методические указания по проведению различных учебных занятий, к курсовому проектированию и другим видам самостоятельной работы	51
8. Материально-техническое обеспечение дисциплины	59
9. Лист изменений (дополнений)	63

1. Цель и задачи освоения дисциплины

Цель освоения дисциплины:

- ознакомление студентов с основными понятиями, структурами, методами и алгоритмами современного программирования;
- ознакомление студентов с различными парадигмами программирования; практически ознакомить студентов с языками программирования высокого уровня Turbo Pascal и C++;
- ознакомление студентов с методами, применяемыми в программировании, известными алгоритмами, возможностью работы с динамической памятью, с конструкциями распределенного и параллельного программирования;
- дать начальный запас сведений о методах и этапах трансляции;
- дать навыки решения задач с использованием современных языков и технологий программирования.

Задачи освоения дисциплины:

- обучить студентов методам и мышлению, характерным для современного программирования;
- сформировать у студентов представление о структурах данных, как о некоторой абстракции, позволяющей описывать объекты реального мира на языке информационных моделей;
- сформировать у студентов представление об общих принципах разработки алгоритмов и анализа их эффективности на примере алгоритмов из различных областей математики, реализуемых в виде компьютерных приложений;
- сформировать представление о современной методологии проектирования и программирования, принципах трансляции и верификации программ;
- сформировать практические навыки разработки алгоритмов, подбора адекватных задач структур данных и их реализации на современных программных средствах.
- создать фундамент освоения новых языков программирования;
- способствовать развитию точного научного мышления, повышению программистской культуры.

2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Языки и методы программирования» относится к перечню дисциплин Блока 1. Часть, формируемая участниками образовательных отношений направления подготовки 02.03.02 Фундаментальная информатика и информационные технологии,

профиль «Проектирование систем искусственного интеллекта» и изучается в 1, 2 семестрах 1 курса.

3. Требования к результатам освоения дисциплины (модуля)

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ОПОП ВО по данному направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии (уровень бакалавриата):

профессиональных (ПКС):

Коды	Содержание компетенций
ПКС- 3	Способен преподавать математику и информатику в средней школе, специальных учебных заведениях на основе полученного фундаментального образования и научного мировоззрения

В результате освоения дисциплины обучающийся должен:

Знать:

- методы описания синтаксических конструкций языков программирования, классификацию и эволюцию языков программирования;
- системы программирования Turbo Pascal и C++ и иметь практическое представление об их общности и особенностях;
- основные положения современной концепции типа данных;
- основные методы, применяемые в программировании, известные алгоритмы;
- технологии программирования;
- методы тестирования алгоритмов и программ;
- средства повышения надежности алгоритмов и программ;
- методы трансляции, структуру транслятора, стадии трансляции программ;
- жизненный цикл программного обеспечения, работы, выполняемые на каждом из его этапов;
- основные концепции объектно-ориентированного программирования;
- основные концепции распределенного и параллельного программирования.

Уметь:

- использовать формальные способы описания языков программирования;
- выполнять все этапы подготовки надежных и наглядных программ решения разнообразных задач на ЭВМ;
- применять на практике современные концепции типов данных, выбирать представления для данных, имеющих сложную структуру;

══

реализовывать основные методы, применяемые в программировании;
 разрабатывать, обосновывать и документировать нетривиальные программы;
 сопоставлять различные языки программирования.

Владеть:

навыками использования информационных технологий для разработки стратегии и приоритетных направлений социальной политики, долгосрочных прогнозов процессов;
 навыками решения практических задач; математическими и количественными методами решения типовых организационно-управленческих задач;
 навыками работы с математической литературой и литературой по программированию, навыками применения современного математического инструментария для решения задач экономики и информатики.
 навыками применения основных методов физико-математического анализа для решения естественнонаучных задач;
 обработки и интерпретирования результатов эксперимента
 средствами компьютерной техники и информационных технологий, приемами навигации по файловой структуре компьютера и управления ее файлами;
 технологией создания научно-технической документации различной сложности с помощью текстового процессора;
 технологией поиска и обмена информацией в глобальных и локальных компьютерных сетях.

4. Содержание и структура дисциплины

Таблица 1. Содержание дисциплины «Языки и методы программирования», перечень оценочных средств и контролируемых компетенций

№ п/п	Наименование раздела	Содержание раздела	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1	2	3	4	5
Раздел I. Языки программирования				
1.	Введение. Основные понятия языков программирования. Среды программирования	Исторический очерк развития языков программирования (ЯП). Основные конструкции языков программирования и их реализация. Методы разработки алгоритмов и программ. Технология программирования.	ПКС- 3	Домашнее задание (ДЗ), Контрольная работа (КР), Коллоквиум (К), Тестирование (Т), Рубежный контроль (РК)

2.	<p>Базисные типы данных и операторный базис в процедурных языках программирования (на примере языка Turbo Pascal 7.0). Методы программирования с использованием статических структур данных.</p>	<p>Данные и операторы. Классификация данных. Классификация операторов. Атрибуты данных и средства их описания. Характеристики, связанные с типом. Структура Pascal-программы. Базисные типы данных. Описания. Оператор присваивания. Организация ввода и вывода данных. Понятие исключительной ситуации. Операторы составной, условный, выбора. Числовые массивы. Блочная структура программы. Процедуры без параметров. Области действия описаний. Правила видимости. Понятие цикла. Организация циклов с известным числом повторений. Кратные циклы с известным числом повторений. Организация циклов с неопределенным числом повторений. Организация прерывания циклов. Кратные циклы с неопределенным числом повторений. Комбинация циклов с определенным и неопределенным числом повторений. Порядковые и вещественные типы. Структурированные типы – массивы, записи, множества, перечислимый тип. Типизированные константы. Строковый и символьный типы. Совместимость и преобразование типов. Комбинированный тип данных. Оператор присоединения. Реализация методов упорядочения, поиска, выбора и слияния с использованием статических структур данных.</p>	ПКС- 3	ДЗ, КР, К, Т, РК
----	--	---	--------	------------------

3.	Процедуры и функции.	Структура процедур. Область действия идентификаторов при использовании процедур. Классификация способов передачи параметров. Передача параметров в Turbo Pascal. Структура функций. Область действия идентификаторов при использовании функций. Передача параметров в Turbo Pascal. Побочный эффект функций. Процедурные директивы near и far, forward, external и др.	ПКС- 3	ДЗ, КР, К, Т, РК
4.	Рекурсивные методы в программировании.	Понятие рекурсии. Рекурсивный спуск. Рекурсивный возврат. Формы рекурсивных процедур. Перекрестная рекурсия.	ПКС- 3	ДЗ, КР, К, Т, РК
5.	Понятие файла. Методы упорядочения Файлов.	Файлы. Доступ к файлам. Процедуры и функции для работы с файлами. Текстовые файлы. Типизированные и не типизированные файлы.	ПКС- 3	ДЗ, КР, К, Т, РК
6.	Понятие модуля. Стандартные библиотечные модули CRT и GRAPH. Пользовательские модули.	Модули. Принципы модульного программирования. Преимущества использования модулей. Структура модулей. Заголовок модуля и связь модулей друг с другом. Интерфейсная часть. Исполняемая часть. Иницирующая часть. Компиляция модулей. Отличие сферы действия переменных в модуле и процедуре. Циклические ссылки модулей. Утилита TpmOver. Стандартные модули. Стандартные библиотечные модули. Модуль CRT. Установка цвета и фона. Управление яркостью. Управление курсором. Управление звуком. Управление клавиатурой. Текстовые окна. Стандартный библиотечный модуль GRAPH. Построение изображений с помощью модуля GRAPH. Вывод текста и численных данных в графическом режиме. Графические окна. Сохранение образов в памяти и восстановление на экране. Управление страницами.	ПКС- 3	ДЗ, КР, К, Т, РК

7.	Несвязанные динамические структуры данных	Динамическая память. Адреса и указатели. Объявление указателей. Выделение и освобождение динамической памяти. Использование указателей. Несвязанные динамические данные.	ПКС- 3	ДЗ, КР, К, Т, РК
8.	Связанные динамические структуры данных. Методы программирования с использованием связанных динамических структур данных.	Процедуры и функции для работы с динамической памятью. Администратор кучи. Сложные структуры данных. Организация взаимосвязей в связанных динамических данных. Очереди. Набор базисных операций над очередями. Дисциплина обработки очередей. Отображение типа очередей на типы данных языка Turbo Pascal и реализация на языке Turbo Pascal его базисных операций. Стеки. Дисциплины обслуживания стеков. Занесение элемента в стек. Выбор элемента из стека. Отображение типа стеков на типы данных языка Turbo Pascal и реализация его базисных операций. Списки. Виды списков. Дисциплина обработки списков. Отображение типа списков на типы данных языка Turbo Pascal и реализация его базисных операций. Двухнаправленные и кольцевые списки. Вставка, удаление и поиск элементов в двухнаправленных и кольцевых списках. Графы и оргграфы. Отображение типов графов и оргграфов на типы данных языка Turbo Pascal и реализация их базисных операций. Тип вершинных графов.	ПКС- 3	ДЗ, КР, К, Т, РК
Раздел II. Методы программирования				

1.	Понятие и методы объектно-ориентированного программирования	<p>Понятие объектно-ориентированного анализа. Связь объектного подхода с основными понятиями языков программирования.</p> <p>Недостатки традиционных ЯП с точки зрения объектного подхода.</p> <p>Разновидности декомпозиции – алгоритмическая и объектно-ориентированная. Наследование в ЯП. Понятия и примеры наследования.</p> <p>Наследование и полиморфизм. Объявление объектов.</p> <p>Конструкторы и деструкторы. Виртуальные методы.</p> <p>Абстрактные методы.</p> <p>Инкапсуляция.</p> <p>Другие парадигмы программирования: функциональное и логическое программирование.</p>	ПКС- 3	ДЗ, КР, К, Т, РК
2.	Операторный базис языка программирования C++	<p>Алфавит. Литералы. Типы данных. Операции. Условная операция. Простое присваивание. Составное присваивание.</p> <p>Операторы: условный, цикла с предусловием, цикла с постусловием, пошагового цикла, разрыва, продолжения, возврата. Реализация ввода/вывода. Структура программы. Объявление и определение функции.</p> <p>Встраиваемые функции.</p> <p>Директивы препроцессора.</p> <p>Массивы: объявления, индексация, инициализация, доступ к элементам, ввод/вывод массивов. Поточковый ввод/вывод. Форматирование.</p> <p>Функции: объявление и определение, глобальные переменные, возвращаемое значение, параметры функций, перегрузка функций, шаблоны функций. Функции стандартной библиотеки.</p>	ПКС- 3	ДЗ, КР, К, Т, РК

3.	Методы трансляции	Основные этапы трансляции и основные понятия лексического анализа. Организация таблиц компилятора. Обратная польская запись (ОПЗ) арифметических выражений. Лексический анализ для выражений в ОПЗ. Синтаксический анализ. Грамматика для арифметических выражений.	ПКС- 3	ДЗ, КР, К, Т, РК
4.	Основные концепции и методы параллельного и распределенного программирования	Параллельное программирование. Распределенное программирование	ПКС- 3	ДЗ, КР, К, Т, РК

Общая трудоёмкость дисциплины составляет 12 зачётных единиц (432 часа).

Таблица 1. Структура дисциплины «Языки и методы программирования»

Вид работы	Трудоёмкость часов / зачётных единиц		
	1 семестр	2 семестр	всего
Общая трудоёмкость (в часах)	180	252	432
Контактная работа (в часах):	51	90	141
<i>Лекционные занятия (Л)</i>			
<i>Практические занятия (ПЗ)</i>	17	36	53
<i>Семинарские занятия (СЗ)</i>			
<i>Лабораторные работы (ЛР)</i>	34	54	88
Самостоятельная работа (в часах), в том числе контактная (внеаудиторная)	102	135	237
Расчетно-графическое задание			
Реферат (Р)			
Эссе (Э)			
Контрольная работа (К)			
Самостоятельное изучение разделов	102	135	237
Курсовой проект (КП), курсовая работа (КР)			
Подготовка и прохождение промежуточной аттестации	27	27	54
Вид промежуточной аттестации	Экзамен	Экзамен	Экзамен

Таблица 2. Лекционные занятия

№ п/п	Тема
1.	Не предусмотрены

Таблица 4. Практические занятия (семинарские занятия)

№ п/п	Тема
1.	Введение. Основные понятия языков программирования. Среды программирования
2.	Базисные типы данных и операторный базис в процедурных языках программирования. Методы программирования с использованием статических структур данных.
3.	Процедуры и функции.
4.	Рекурсивные методы в программировании.
5.	Понятие файла. Методы упорядочения файлов.
6.	Понятие модуля. Стандартные библиотечные модули CRT и GRAPH. Пользовательские модули.
7.	Несвязанные динамические структуры данных.
8.	Связанные динамические структуры данных. Методы программирования с использованием связанных динамических структур данных
9.	Понятие и методы объектно-ориентированного программирования
10.	Операторный базис языка программирования C++
11.	Методы трансляции
12.	Основные концепции и методы параллельного и распределенного программирования

Таблица 5. Лабораторные работы

№ п/п	Наименование темы
1.	Введение. Основные понятия языков программирования. Среды программирования
2.	Базисные типы данных и операторный базис в процедурных языках программирования. Методы программирования с использованием статических структур данных.
3.	Процедуры и функции.
4.	Рекурсивные методы в программировании.
5.	Понятие файла. Методы упорядочения файлов.
6.	Понятие модуля. Стандартные библиотечные модули CRT и GRAPH. Пользовательские модули.
7.	Несвязанные динамические структуры данных.
8.	Связанные динамические структуры данных. Методы программирования с использованием связанных динамических структур данных
9.	Понятие и методы объектно-ориентированного программирования
10.	Операторный базис языка программирования C++
11.	Методы трансляции
12.	Основные концепции и методы параллельного и распределенного программирования

Таблица 6. Самостоятельное изучение разделов дисциплины (модуля)

№ п/п	Вопросы, выносимые на самостоятельное изучение
1.	Исторический очерк развития языков программирования (ЯП). Основные конструкции языков программирования и их реализация. Методы разработки алгоритмов и программ. Технология программирования.

2.	<p>Данные и операторы. Классификация данных. Классификация операторов. Атрибуты данных и средства их описания. Характеристики, связанные с типом. Структура Pascal-программы. Базисные типы данных. Описания. Оператор присваивания. Организация ввода и вывода данных. Понятие исключительной ситуации. Операторы составной, условный, выбора. Числовые массивы. Блочная структура программы. Процедуры без параметров. Области действия описаний. Правила видимости. Понятие цикла. Организация циклов с известным числом повторений. Кратные циклы с известным числом повторений. Организация циклов с неопределенным числом повторений. Организация прерывания циклов. Кратные циклы с неопределенным числом повторений. Комбинация циклов с определенным и неопределенным числом повторений. Порядковые и вещественные типы. Структурированные типы – массивы, записи, множества, перечислимый тип. Типизированные константы. Строковый и символьный типы. Совместимость и преобразование типов. Комбинированный тип данных. Оператор присоединения. Реализация методов упорядочения, поиска, выбора и слияния с использованием статических структур данных.</p>
3.	<p>Структура процедур. Область действия идентификаторов при использовании процедур. Классификация способов передачи параметров. Передача параметров в Turbo Pascal. Структура функций. Область действия идентификаторов при использовании функций. Передача параметров в Turbo Pascal. Побочный эффект функций. Процедурные директивы near и far, forward, external и др.</p>
4.	<p>Понятие рекурсии. Рекурсивный спуск. Рекурсивный возврат. Формы рекурсивных процедур. Перекрестная рекурсия.</p>
5.	<p>Файлы. Доступ к файлам. Процедуры и функции для работы с файлами. Текстовые файлы. Типизированные и нетипизированные файлы.</p>
6.	<p>Модули. Принципы модульного программирования. Преимущества использования модулей. Структура модулей. Заголовок модуля и связь модулей друг с другом. Интерфейсная часть. Исполняемая часть. Иницилирующая часть. Компиляция модулей. Отличие сферы действия переменных в модуле и процедуре. Циклические ссылки модулей. Утилита ТриМover. Стандартные модули. Стандартные библиотечные модули. Модуль CRT. Установка цвета и фона. Управление яркостью. Управление курсором. Управление звуком. Управление клавиатурой. Текстовые окна. Стандартный библиотечный модуль GRAPH. Построение изображений с помощью модуля GRAPH. Вывод текста и численных данных в графическом режиме. Графические окна. Сохранение образов в памяти и восстановление на экране. Управление страницами.</p>
7.	<p>Динамическая память. Адреса и указатели. Объявление указателей. Выделение и освобождение динамической памяти. Использование указателей. Несвязанные динамические данные.</p>

8.	<p>Процедуры и функции для работы с динамической памятью. Администратор кучи. Сложные структуры данных. Организация взаимосвязей в связанных динамических данных.</p> <p>Очереди. Набор базисных операций над очередями. Дисциплина обработки очередей. Отображение типа очередей на типы данных языка Turbo Pascal и реализация на языке Turbo Pascal его базисных операций.</p> <p>Стеки. Дисциплины обслуживания стеков. Занесение элемента в стек. Выбор элемента из стека. Отображение типа стеков на типы данных языка Turbo Pascal и реализация его базисных операций.</p> <p>Списки. Виды списков. Дисциплина обработки списков. Отображение типа списков на типы данных языка Turbo Pascal и реализация его базисных операций.</p> <p>Двунаправленные и кольцевые списки. Вставка, удаление и поиск элементов в двунаправленных и кольцевых списках.</p> <p>Графы и оргграфы. Отображение типов графов и оргграфов на типы данных языка Turbo Pascal и реализация их базисных операций. Тип вершинных графов.</p>
9.	<p>Понятие объектно-ориентированного анализа. Связь объектного подхода с основными понятиями языков программирования. Недостатки традиционных ЯП с точки зрения объектного подхода.</p> <p>Разновидности декомпозиции – алгоритмическая и объектно-ориентированная.</p> <p>Наследование в ЯП. Понятия и примеры наследования.</p> <p>Наследование и полиморфизм. Объявление объектов. Конструкторы и деструкторы.</p> <p>Виртуальные методы. Абстрактные методы. Инкапсуляция.</p> <p>Другие парадигмы программирования: функциональное и логическое программирование.</p>
10.	<p>Алфавит. Литералы. Типы данных. Операции. Условная операция. Простое присваивание. Составное присваивание. Операторы: условный, цикла с предусловием, цикла с постусловием, пошагового цикла, разрыва, продолжения, возврата. Реализация ввода/вывода. Структура программы. Объявление и определение функции. Встраиваемые функции. Директивы препроцессора.</p> <p>Массивы: объявления, индексация, инициализация, доступ к элементам, ввод/вывод массивов. Поточковый ввод/вывод. Форматирование. Функции: объявление и определение, глобальные переменные, возвращаемое значение, параметры функций, перегрузка функций, шаблоны функций. Функции стандартной библиотеки.</p>
11.	<p>Основные этапы трансляции и основные понятия лексического анализа.</p> <p>Организация таблиц компилятора.</p> <p>Обратная польская запись (ОПЗ) арифметических выражений. Лексический анализ для выражений в ОПЗ. Синтаксический анализ.</p> <p>Грамматика для арифметических выражений.</p>
12.	<p>Параллельное программирование. Распределенное программирование.</p>

5. Оценочные материалы для текущего и рубежного контроля успеваемости и промежуточной аттестации

Конечными результатами освоения программы дисциплины являются сформированные когнитивные дескрипторы «знать», «уметь», «владеть», расписанные по отдельным компетенциям. Формирование этих дескрипторов происходит в течение всего семестра по этапам в рамках различного вида занятий и самостоятельной работы.

В ходе изучения дисциплины предусматриваются *текущий, рубежный контроль и промежуточная аттестация.*

5.1. Оценочные материалы для текущего контроля

Текущий контроль знаний, умений и владений по дисциплине осуществляется в форме устного или письменного опроса на лекционных и практических занятиях, а также в ходе проведения самостоятельной работы студентов.

Цель текущего контроля – оценка результатов работы в семестре и обеспечение своевременной обратной связи для коррекции обучения, активизации самостоятельной работы обучающегося. Объектом текущего контроля являются конкретизированные результаты обучения (учебные достижения) по дисциплине.

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины «Языки и методы программирования» и включает: ответы на теоретические вопросы на практическом занятии, решение практических задач и выполнение заданий на практическом занятии, самостоятельное выполнение индивидуальных домашних заданий (например, решение задач) с отчетом (защитой) в установленный срок, написание курсовой работы.

Оценка качества подготовки на основании выполненных заданий ведется преподавателем (с обсуждением результатов), баллы начисляются в зависимости от сложности задания.

5.1.1. Вопросы по темам дисциплины «Языки и методы программирования» (контролируемые компетенции ПКС-2)

Тема 1. Введение. Основные понятия языков программирования. Среды программирования

1. Исторический очерк развития языков программирования (ЯП).
2. Основные конструкции языков программирования и их реализация.
3. Методы разработки алгоритмов и программ.
4. Технология программирования.

Тема 2. Базисные типы данных и операторный базис в процедурных языках программирования. Методы программирования с использованием статических структур данных

5. Данные, операторы и их классификация
 6. Атрибуты данных и средства их описания. Характеристики, связанные с типом.
 7. Структура Pascal-программы.
 8. Базисные типы данных.
-

9. Описания. Оператор присваивания.
10. Организация ввода и вывода данных.
11. Понятие исключительной ситуации.
12. Операторы составной, условный, выбора.
13. Числовые массивы.
14. Блочная структура программы.
15. Процедуры без параметров.
16. Области действия описаний.
17. Правила видимости.
18. Понятие цикла.
19. Порядковые и вещественные типы.
20. Структурированные типы – массивы, записи, множества, перечислимый тип.
21. Типизированные константы.
22. Строковый и символьный типы.
23. Совместимость и преобразование типов.
24. Комбинированный тип данных. Оператор присоединения.
25. Реализация методов упорядочения, поиска, выбора и слияния с использованием статических структур данных.

Тема 3. Процедуры и функции.

1. Структура процедур.
2. Область действия идентификаторов при использовании процедур.
3. Классификация способов передачи параметров.
4. Передача параметров в Turbo Pascal.
5. Структура функций.
6. Область действия идентификаторов при использовании функций.
7. Побочный эффект функций.
8. Процедурные директивы near и far, forward, external и др.

Тема 4. Рекурсивные методы в программировании.

1. На чем основан рекурсивный метод программирования?
2. В чем состоит мощьность рекурсивных определений?
3. В чем преимущества и недостатки использования рекурсивных процедур по сравнению с нерекурсивными?
4. Что называется «глубиной рекурсии»?
5. Какой элемент является обязательным в рекурсивном определении?

==

6. К каким последствиям приводит «рекурсивное заикливание»?
7. Какое условие должно обязательно присутствовать в любой рекурсивной процедуре?
8. Что такое явная и косвенная рекурсии?
9. Дайте рекурсивное определение целой степени числа N.
10. Почему простой рекурсивный алгоритм для вычисления чисел Фибоначчи неэффективен?
11. Можно ли утверждать, что нерекурсивный алгоритм для вычисления факториала намного эффективнее рекурсивного?

Тема 5. Понятие файла. Методы упорядочения файлов

1. В каких случаях удобно использовать файлы?
2. Где хранятся файлы?
3. В чем состоит сходство и различие между массивом и файлом?
4. По каким признакам классифицируются файлы в Turbo Pascal?
5. Как в разделе типов задается файловых тип?
6. Как подразделяются файлы по видам доступа к его компонентам?
7. Каким образом описываются переменные файловых типов?
8. Что необходимо выполнить для открытия файла?
9. Какие процедуры предназначены для открытия файлов и как они работают?
10. Для чего предназначена процедура Close?
11. Как осуществляется доступ к компонентам файла?
12. Какие операции определены над файлами?
13. Какие процедуры предназначены для работы с каталогами?
14. Какие действия выполняют процедуры Rename и Erase?
15. Каких типов допускаются описания типизированных файлов?
16. Как нумеруются элементы типизированных файлов?
17. По каким правилам выполняется чтение из типизированных файлов?
18. Какой формат имеет процедура Write для типизированных файлов?
19. Какие процедуры и функции предназначены для прямого доступа к элементам типизированных файлов?
20. В чем состоят особенности текстовых файлов?
21. В чем отличие пользовательских текстовых файлов от файлов Input и Output?
22. Определить тип стандартных файлов Input и Output. Нужно ли их описывать в программе? Нужно ли эти файлы открывать на запись или на чтение?
23. В чем отличие текстового файла от file of Char ?

24. Как объявляется нетипизированный файл?
25. В чем отличие нетипизированных файлов от типизированных?
26. Какие процедуры и функции используются для работы с нетипизированными файлами?

Тема 6. Понятие модуля. Стандартные библиотечные модули CRT и GRAPH.

Пользовательские модули.

1. Что собой представляет модуль?
2. Какие имеются стандартные модули в составе систем программирования Turbo Pascal 6.0 и 7.0?
3. Кратко охарактеризуйте стандартные модули System, Crt, Dos, Graph, Overlay, Printer.
4. В чем состоит основной принцип модульного программирования?
5. Какие преимущества обеспечивает использование модулей?
6. Какие модули подключаются по умолчанию?
7. В каком разделе программы указывается подключение модулей?
8. Какие существуют особенности при разработке собственных модулей?
9. В чем состоит отличие сферы действия переменных в модуле и процедуре?
10. Из каких разделов состоит модуль?
11. Что описывается в разделе Interface?
12. Что описывается в разделе Implementation?
13. Что описывается в разделе инициализации?
14. Обязателен ли раздел инициализации?
15. В чем заключается разница описаний заголовков процедур (функций)
16. в разделе Interface и в разделе Implementation?
17. Опишите механизм использования циклических ссылок модулей.
18. Как компилируется модуль?
19. В чем разница режимов компиляции Compile/Make и Compile/Build?
20. В чем заключается преимущество использования модулей при написании больших программ?

Тема 7. Несвязанные динамические структуры данных

1. На какие группы можно разделить используемые в программировании типы данных?
2. В чем состоит основное отличие между данными статической и динамической структуры?
3. Какие данные относятся к данным статической структуры?
4. Какие данные относятся к однородным?

==

5. Какие данные относятся к неоднородным?
6. По каким признакам могут быть классифицированы массивы?
7. Какие неоднородные структуры данных реализованы в Turbo Pascal?
8. Какие данные относятся к данным динамической структуры?
9. Какие существуют разновидности связанных динамических данных?
10. В чем особенность объявления данных динамической структуры?
11. С помощью каких процедур происходит распределение памяти под динамические переменные?
12. Что означает состояние nil указателя?

Тема 8. Связанные динамические структуры данных. Методы программирования с использованием связанных динамических структур данных

1. Какие действия выполняют процедуры New и Dispose?
2. В чем выражаются динамические свойства несвязанных динамических данных?
3. В чем сходство и различие между линейными и кольцевыми списками?
4. Что представляет собой очередь как структура данных?
5. Что представляет собой стек как структура данных?
6. Что представляет собой пирамида как структура данных и в чем ее отличие от дерева?
7. Что требуется для создания связанных динамических структур данных?
8. Сколько указателей требуется для работы с очередью?
9. Какие действия необходимо выполнить для работы с очередью?
10. Как добавить или удалить элемент очереди?
11. Какие действия необходимо выполнить для работы со стеком?
12. Как добавить или удалить элемент стека?
13. Сколько указателей требуется для работы со стеком?

Тема 9. Понятие и методы объектно-ориентированного программирования

1. В чем суть объектно-ориентированной методологии программирования (ООП) ?
2. Каково назначение ООП?
3. Дайте определение понятию «объект» в ООП.
4. Опишите в общем виде тип object в языке Turbo Pascal.
5. Опишите разновидности декомпозиции: алгоритмическую и объектно-ориентированную. Приведите примеры каждого из видов декомпозиции.
6. В чем суть принципов инкапсуляции, наследования и полиморфизма?
7. Что такое «абстрактные типы»? На каких уровнях ОО-иерархии они обычно размещаются?

8. Опишите группы иерархии объектов программы-редактора.
9. Дайте описание типов «Связь элементов» и «Элемент строки».
10. Опишите назначение ключевых слов `private` и `inherited`.
11. Какие методы называются виртуальными? Каково основное назначение виртуальных методов?
12. Как объявляются виртуальные методы?
13. Что такое конструкторы и деструкторы?
14. Как формируется таблица виртуальных методов?
15. Опишите две основные возможности модификации программ при использовании виртуальных методов.

Тема 10. Операторный базис языка программирования C++

1. Алфавит.
2. Литералы.
3. Типы данных.
4. Операции.
5. Условная операция.
6. Простое присваивание. Составное присваивание.
7. Операторы: условный, цикла с предусловием, цикла с постусловием, пошагового цикла, разрыва, продолжения, возврата.
8. Реализация ввода/вывода.
9. Структура программы.
10. Объявление и определение функции. Встраиваемые функции.
11. Директивы препроцессора.
12. Массивы: объявления, индексация, инициализация, доступ к элементам, ввод/вывод массивов.
13. Поточковый ввод/вывод. Форматирование.
14. Функции: объявление и определение, глобальные переменные, возвращаемое значение, параметры функций, перегрузка функций, шаблоны функций.
15. Функции стандартной библиотеки.

Тема 11. Методы трансляции

1. Основные этапы трансляции и основные понятия лексического анализа.
2. Организация таблиц компилятора.
3. Обратная польская запись (ОПЗ) арифметических выражений.
4. Лексический анализ для выражений в ОПЗ.

══

5. Синтаксический анализ.
6. Грамматика для арифметических выражений.

Тема 12. Основные концепции и методы параллельного и распределенного программирования

1. Параллелизм как основа высокопроизводительных вычислений. Уровни параллелизма.
2. Симметричные мультипроцессорные системы.
3. Системы с массовым параллелизмом.
4. Распределенные системы: основные понятия и проблемы организации.
5. Программный параллелизм. Зависимости по данным, по управлению и по ресурсам.
6. Аппаратный параллелизм: идеальный параллелизм, конвейерный параллелизм, синхронный и асинхронный параллелизм.
7. Проблема когерентности памяти при организации параллельных вычислений.
8. Мультипроцессорная когерентность кэш-памяти.
9. Метрики параллельных вычислений: профиль параллелизма программы; ускорение и эффективность; коэффициент использования (избыточность) параллелизма и качество.
10. Закон Амдала и его следствия.
11. Закон Густафсона для масштабируемых задач.
12. Модели параллельных вычислений.
13. Классификация параллельных вычислительных систем (классификация Флинна).
14. Кластерные вычислительные системы. Основные свойства кластерных систем.
15. Классификация архитектур кластерных систем. Топологии кластеров. Требования к аппаратному обеспечению кластеров.

Критерии формирования оценок (оценивания) устного опроса

Устный опрос является одним из основных способов учёта знаний обучающегося по дисциплине. В результате устного опроса знания, обучающегося оцениваются по следующей шкале:

Количество баллов	Критерии оценивания
-------------------	---------------------

5	Обучающийся - полно излагает изученный материал, знает все формулы, применяемые методы и их точность; - понимает материал, может обосновать свои суждения, применить знания при решении практических задач и лабораторных заданий для самостоятельного выполнения; - излагает материал последовательно и правильно с точки зрения норм литературного языка.
4	Обучающийся даёт ответ, удовлетворяющий тем же требованиям, что и для «5» баллов, но допускает несущественные ошибки, которые сам же исправляет, и некоторые недочёты в последовательности и оформлении излагаемого материала.
3	Обучающийся обнаруживает знание и понимание основного материала по данной теме, но: - излагает материал неполно и допускает неточности в определении понятий, знаний методов, их точности; - не умеет достаточно глубоко и доказательно обосновать свои суждения и применять методы; - излагает материал непоследовательно, допускает ошибки.
2	Обучающийся обнаруживает неполное незнание некоторой части раздела изучаемого материала, допускает ошибки в формулировке и формулах, при оценке точности методов.
1	Обучающийся обнаруживает незнание некоторой части раздела изучаемого материала, допускает существенные ошибки в формулировке и формулах, при оценке точности методов.
0	Обучающийся обнаруживает незнание большей части раздела изучаемого материала, допускает ошибки в формулировке и формулах, при оценке точности методов.

5.1.2 Оценочные материалы для самостоятельной работы обучающегося (типовые задачи) (контролируемые компетенции ПКС-3)

Тема 1. Введение. Основные понятия языков программирования. Среды программирования

1. Организация входа в интегрированную среду
2. Этапы разработки программы
3. Основное меню среды программирования
4. Открытие и сохранение файла в среде программирования
5. Организация поиска нужного фрагмента программы
6. Компиляция и запуск программы в среде программирования
7. Организация работы с точками останова
8. Установка необходимых параметров интегрированной среды
9. Получение справочной информации
10. Полный цикл работ по набору, отладке и исполнению примеров программ с

применением средств пошаговой отладки среды программирования.

Тема 2. Базисные типы данных и операторный базис в процедурных языках программирования. Методы программирования с использованием статических структур данных

1. Вычислите длину окружности, площадь круга и объём шара одного и того же заданного радиуса.
2. Определите номер квадранта, в котором находится точка с заданными координатами (x, y) .
3. Подсчитайте число и сумму положительных, число и произведение отрицательных элементов заданного массива $A(N)$.
4. Запишите подряд в массив $A(N)$ элементы заданного массива $B(2N)$, стоящие на чётных местах, а элементы, стоящие на нечетных местах, запишите в массив $C(N)$.
5. Дана матрица $A(N, M)$. Найдите её наибольший элемент и номера строки и столбца, на пересечении которых он находится.
6. Дана матрица $A(N, N)$ и целое P . Преобразуйте матрицу по правилу: строку с номером P сделайте столбцом с номером P , а столбец с номером P сделайте строкой с номером P .
7. Выясните, имеются ли в заданном векторе $A(N)$ два подряд идущих нулевых элемента.
8. Все элементы заданного вектора $A(N)$, начиная с первого по порядку положительного элемента, уменьшите на единицу.
9. В заданной целочисленной матрице $A(N, M)$ выведите на печать индексы первого положительного элемента, кратного заданному числу K . Если таких элементов в матрице нет, то выведите соответствующий текст. Элементы матриц просматривайте слева направо и сверху вниз.
10. Элементы заданной матрицы $A(N, N)$ переписывайте построчно в одномерный массив до тех пор, пока не встретится нулевой элемент.
11. В заданной матрице $A(N, M)$ найдите количество строк, не содержащих отрицательных чисел.
12. Дана матрица $A(N, M)$. Постройте вектор $B(N)$, элементы B_i которого равны единице, если элементы i -й строки образуют упорядоченную по убыванию или по возрастанию последовательность, и нулю во всех остальных случаях.

Тема 3. Процедуры и функции

1. Из заданного множества точек на плоскости выберите две различные точки так, чтобы количества точек, лежащих по разные стороны прямой, проходящей через эти две

==

- точки, различались наименьшим образом.
2. Определите радиус и центр окружности, на которой лежит наибольшее число точек заданного на плоскости множества точек.
 3. Задано множество M точек на плоскости. Определите, верно ли, что для каждой точки $A \in M$ существует точка $B \in M$ ($A \neq B$) такая, что не существует двух точек множества M , лежащих по разные стороны от прямой AB .
 4. Определите радиус и центр такой окружности, проходящей хотя бы через три различные точки заданного множества точек на плоскости, что минимальна разность количеств точек, лежащих внутри и вне окружности.
 5. В множестве точек на плоскости найдите пару точек с максимальным расстоянием между ними.
 6. Расстояние между двумя множествами точек – это расстояние между наиболее близко расположенными точками этих множеств. Найдите расстояние между двумя заданными множествами точек на плоскости.
 7. Многоугольник (не обязательно выпуклый) задан на плоскости перечислением координат вершин в порядке обхода его границы. Определите площадь многоугольника.
 8. Задано множество точек M в трехмерном пространстве. Найдите такую из них, что шар заданного радиуса с центром в этой точке содержит максимальное число точек из M .
 9. Задано множество прямых на плоскости (коэффициентами своих уравнений). Подсчитайте количество точек пересечения этих прямых.
 10. В трехмерном пространстве задано множество материальных точек. Найдите ту из них, которая расположена наиболее близко к центру тяжести этого множества.

Тема 4. Рекурсивные методы в программировании

1. Посредством рекурсивной функции $\text{summ}(s, i, j)$ проверьте, является ли симметричной часть строки s , начинающаяся i -м и заканчивающаяся j -м её элементами.
 2. «Переверните» заданное натуральное число, используя рекурсивную функцию.
 3. Найдите минимальный элемент массива $A(N)$, используя рекурсию.
 4. Найдите наибольший общий делитель двух натуральных чисел. Используйте рекурсию.
 5. Проверьте, является ли заданное натуральное число простым. Используйте рекурсивную функцию или процедуру.
 6. Реализуйте рекурсивный алгоритм построения цепочки из имеющегося набора костей домино.
-

7. Задан набор слов. Рекурсивно постройте из них цепочку максимальной длины таким образом, чтобы символ в конце слова совпадал с символом в начале следующего.
8. Выполните перевод числа из десятичной системы счисления в двоичную, используя рекурсивную процедуру/функцию.
9. Посредством рекурсивной функции определите количество единиц в двоичном представлении натурального числа.
10. Подсчитайте количество различных представлений заданного натурального m в виде суммы не менее двух попарно различных положительных слагаемых. Представления, различающиеся лишь порядком слагаемых, считаются одинаковыми.
11. Напишите рекурсивную программу, которая в заданном дереве по заданному n считает число всех вершин высоты n и печатает список этих вершин (в том или ином порядке).
12. Напишите рекурсивную программу, которая печатает все перестановки чисел $1, \dots, n$ по одному разу.
13. Перечислите все представления положительного целого числа n в виде суммы последовательности невозрастающих целых положительных слагаемых. Используйте рекурсивную процедуру.
14. Напишите рекурсивную программу обхода дерева во внутреннем порядке.

Тема 5. Понятие файла. Методы упорядочения файлов

Задача 1

1. Создайте файл из 50 целых чисел, используя генератор случайных чисел. Выведите эти числа на экран. Выведите на экран все числа из этого файла, оканчивающиеся на 7. Найдите среднее арифметическое всех чисел, входящих в созданный файл. Найдите наименьшее из них.

2. Создайте два упорядоченных файла целых чисел. Объедините их, создав новый упорядоченный файл.

3. Создайте текстовый файл. Подсчитайте количество предложений в нем. Найдите самое длинное слово.

Задача 2

1. Создайте файл из 55 целых чисел, используя генератор случайных чисел. Перепишите все четные числа из этого файла в другой. Выведите оба файла на экран. Найдите в новом файле среднее арифметическое его элементов. Найдите наибольший элемент в новом файле.

2. Используя генератор случайных чисел, создайте два упорядоченных по

возрастанию файла: первый – из четных чисел, второй – из нечетных чисел.

3. Создайте текстовый файл. Подсчитайте количество знаков препинания в нем. Найдите самое короткое слово.

Задача 3

1. Используя генератор случайных чисел, создайте файл из 20 целых чисел, кратных 5. Выведите его на экран. Найдите среднее арифметическое этих чисел. Найдите количество элементов, оканчивающихся нулем. Найдите наименьший из таких элементов.

2. Создайте два упорядоченных по убыванию файла. Объедините их в упорядоченный по убыванию файл.

3. Создайте текстовый файл. Уберите лишние пробелы. Выведите на экран слова, начинающиеся с буквы «А».

Задача 4

1. С помощью датчика случайных чисел создайте файл из 35 целых чисел. Выведите его на экран. Найдите наибольшее из чисел, оканчивающихся на 3. Подсчитайте количество чисел, оканчивающихся на 3. Найдите среднее арифметическое остальных чисел.

2. С помощью датчика случайных чисел создайте два упорядоченных по убыванию файла. Объедините их в упорядоченный по возрастанию файл.

3. Создайте текстовый файл. Определите количество вопросительных предложений и количество слов в первом предложении.

Задача 5

1. С помощью датчика случайных чисел создайте файл из 40 целых чисел. Выведите все четные числа, встречающиеся в созданном файле. Подсчитайте их количество. Найдите их среднее арифметическое. Найдите число, ближайшее к среднему арифметическому.

2. С помощью датчика случайных чисел создайте два упорядоченных файла: один – по убыванию, другой – по возрастанию.

3. Создайте текстовый файл. Зашифруйте его, используя метод обратного чтения.

Задача 6

1. Создайте файл из 20 вещественных чисел, выбранных случайным образом из числового отрезка $[10, 30]$. Выведите его на экран. Найдите наибольший и наименьший элементы в этом файле. Найдите сумму всех элементов файла.

2. С помощью датчика случайных чисел создайте два упорядоченных файла: один – по неубыванию, другой – по невозрастанию. Объедините эти файлы в новый,

==

упорядоченный по убыванию файл.

3. Создайте текстовый файл. Выведите его на экран. Выведите на экран последнее предложение. Определите количество слов в нем.

Задача 7

1. Создайте файл из 30 вещественных чисел, выбранных случайным образом из числового отрезка [5, 8]. Выведите его на экран. Найдите среднее арифметическое его элементов. Найдите все элементы, расстояние которых от среднего арифметического не превышает 1, и запишите их в новый файл.

2. С помощью датчика случайных чисел создайте два упорядоченных по возрастанию файла. Объедините их в упорядоченный по убыванию файл.

3. Создайте текстовый файл. Выведите его на экран. Определите, есть ли в тексте слово «утро».

Задача 8

1. Создайте файл из 25 вещественных чисел, выбранных случайным образом из числового отрезка [5, 7]. Найдите наибольший и наименьший элементы. Найдите среднее арифметическое первых десяти элементов.

2. Создайте два файла. Первый – по закону $x_i = 1/i$, второй – $y_i = 1/(i+1)$, где $i = 1, \dots, 20$. Объедините их в упорядоченный по возрастанию файл.

3. Создайте текстовый файл. Выведите его на экран. Поменяйте местами первое и последнее слово.

Задача 9

1. Создайте файл из 50 вещественных чисел, выбранных случайным образом из числового отрезка [1, 4]. Выведите файл на экран. Найдите наибольший элемент. Найдите среднее арифметическое последних 15 чисел.

2. Создайте два файла. Первый – по закону $x_i = i^2$, где $i = 1, \dots, 10$, а второй – по закону $y_k = k^2 + 1$, где $k = 1, \dots, 15$. Создайте новый файл, упорядоченный по убыванию, путем слияния двух ранее созданных.

3. Создайте текстовый файл. Выведите его на экран. Подсчитайте, сколько в этом файле слов, оканчивающихся на букву «д».

Задача 10

1. Создайте файл из 25 вещественных чисел, выбранных случайным образом из числового отрезка [1, 8]. Выведите файл на экран. Найдите наименьший элемент. Найдите среднее арифметическое всех элементов. Выведите на экран все элементы, которые отличаются от среднего арифметического не более чем на 0,5.

==

2. Создайте два файла. Первый – по правилу $x_i = 3i + 1$, где $i = 5, \dots, 15$, второй – по правилу $y_k = 2k - 1$, где $k = 4, \dots, 18$. Получите новый файл путем слияния двух созданных. Упорядочьте его элементы по возрастанию.

3. Создайте текстовый файл. Выведите его на экран. Выведите на экран самое длинное предложение файла.

Тема 6. Понятие модуля. Стандартные библиотечные модули CRT и GRAPH. Пользовательские модули

Разработайте тест модуля и использующую ее программу для решения следующих задач:

Задача 1

1. Создайте файл из 30 вещественных чисел, выбранных случайным образом из числового отрезка $[1, 5]$. Найдите его наибольший элемент. Найдите среднее арифметическое 18 последних чисел, среднее арифметическое всех чисел.

2. Создайте два файла. Первый – по правилу $x_i = 2i + 1$, где $i = 1, \dots, 9$, второй – по правилу $y_k = 1 - 2k$, где $k = 1, \dots, 20$. Получите новый файл путем слияния двух созданных. Упорядочьте его элементы по возрастанию.

3. Создайте текстовый файл. Выведите его на экран. Выведите на экран второе и предпоследнее предложения.

Задача 2

1. Используя датчик случайных чисел, создайте файл из 28 целых чисел, принадлежащих числовому отрезку $[30, 100]$. Выведите его на экран. Найдите среднее арифметическое всех нечетных чисел. Найдите наименьшее среди четных чисел.

2. Создайте два файла. Первый – по правилу $x_i = (1 - 2i)/i$, где $i = 1, \dots, 20$, второй – по правилу $y_k = 1/k$, где $k = 1, \dots, 10$. Получите новый файл путем слияния двух созданных. Упорядочьте его элементы по убыванию.

3. Создайте текстовый файл. Выведите его на экран. Выведите на экран все предложения, начинающиеся с буквы «Ц».

Задача 3

1. Дан файл f , компоненты которого являются действительными числами. Найдите:
а) сумму компонент файла f ; б) произведение компонент файла f ; в) сумму квадратов компонент файла f ; г) модуль суммы и квадрат произведения компонент файла f ; д) последнюю компоненту файла.

2. Даны символьные файлы f и g . Определите, совпадают ли компоненты файла f с компонентами файла g . Если нет, то получите номер первой компоненты, в которой

==

файлы f и g различаются между собой.

3. Дан текстовый файл f . Получите все его строки, содержащие более 60 символов.

Задача 4

1. Дан файл f , компоненты которого являются действительными числами. Найдите: а) наибольшее из значений компонент; б) наименьшее из значений компонент с четными номерами; в) наибольшее из значений модулей компонент с нечетными номерами; г) сумму наибольшего и наименьшего из значений компонент; д) разность первой и последней компонент файла.

2. Дан символьный файл f . Получите его перевернутую копию в файле g .

3. Дан текстовый файл f . Перепишите в файл g все компоненты файла f с заменой в них символа «0» на символ «1» и наоборот.

Задача 5

1. Дан файл f , компоненты которого являются целыми числами. Найдите: а) количество четных чисел среди компонент; б) количество удвоенных нечетных чисел среди компонент; в) количество квадратов нечетных чисел среди компонент.

2. Даны символьные файлы $f1$ и $f2$. Перепишите с сохранением порядка следования компоненты файла $f1$ в файл $f2$, а компоненты файла $f2$ – в файл $f1$. Используйте вспомогательный файл h .

3. Дан текстовый файл f . Запишите в перевернутом виде строки файла f в файл g . Порядок строк в файле g должен: а) совпадать с порядком исходных строк в файле f ; б) быть обратным по отношению к порядку строк исходного файла.

Задача 6

1. Дан файл f , компоненты которого являются целыми числами. Получите в файле g все компоненты файла f :

- а) являющиеся четными числами;
- б) делящиеся на 3 и не делящиеся на 7;
- в) являющиеся точными квадратами.

2. Дан символьный файл f . Добавьте в его конец символы «e», «n», «d» (если это необходимо, используйте дополнительный файл g).

3. Дан текстовый файл f . Перепишите компоненты файла f в файл g , вставляя в начало каждой строки по одному пробелу. Порядок компонент должен быть сохранен.

Задача 7

1. Дан файл f , компоненты которого являются целыми числами. Запишите в файл g все четные числа файла f , а в файл h – все нечетные. Порядок следования чисел

==

сохраняется.

2. Дан символьный файл f . Подсчитайте число вхождений в файл буквосочетаний «ab». Определите, входит ли в файл буквосочетание «abcdefgh». Подсчитайте число вхождений в файл каждой из букв «a», «b», «c», «d», «e», «f» и выведите результат в виде таблицы

a – N	b – N	c – N
d – N	e – N	f – N

3. Дан текстовый файл f . Получите самую длинную строку файла. Если в файле имеется несколько строк с наибольшей длиной, то получите одну из них.

Задача 8

1. Дан файл f , компоненты которого являются целыми числами. Ни одна из компонент файла не равна нулю. Файл f содержит столько же отрицательных чисел, сколько и положительных. Используя вспомогательный файл h , перепишите компоненты файла f в файл g так, чтобы в файле g :

а) не было двух соседних чисел с одним знаком;

б) сначала шли положительные, потом отрицательные числа;

в) числа шли в следующем порядке: два положительных, два отрицательных, два положительных, два отрицательных и т.д. (предполагается, что число компонент в файле f делится на 4).

2. Дан символьный файл f . Запишите в файл g компоненты файла f в обратном порядке.

3. Дан текстовый файл f . Исключите пробелы, стоящие в концах его строк. Результат поместите в файл f .

Задача 9

1. Дан файл f , компоненты которого являются целыми числами. Число компонент файла делится на 100. Запишите в файл g наибольшее значение первых ста компонент файла f , затем следующих ста компонент и т.д.

2. Даны символьные файлы f и g . Запишите в файл h все начальные совпадающие компоненты файлов f и g .

3. Даны текстовый файл и строка s . Получите все строки файла f , содержащие в качестве фрагмента строку s .

Задача 10

1. Дан файл f , компоненты которого являются целыми числами. Получите файл g , образованный из файла f исключением повторных вхождений одного и того же числа.

==

2. Сведения об автомобиле состоят из его марки, номера и фамилии владельца. Дан файл f, содержащий сведения о нескольких автомобилях. Найдите:

- а) фамилии владельцев и номера автомобилей данной марки;
- б) количество автомобилей каждой марки.

3. Даны два текстовых файла f и g. Определите, совпадают ли компоненты файла f с компонентами файла g. Если нет, то получите номер первой строки и позицию первого символа в этой строке, в которых файлы f и g отличаются между собой.

Тема 7. Несвязанные динамические структуры данных

1. Сведения о 10 персонах (дата рождения: число, день недели, месяц, год; место жительства: город, улица, номер дома, номер квартиры);
2. Семинар (предмет, преподаватель, номер группы, день недели, часы занятий, аудитория);
3. Сведения о компьютерах (название; объем памяти; быстродействие, стоимость в рублях и в долларах; фирма-поставщик; тип монитора и дисководов);
4. Бланк требования на книгу (сведения о книге: шифр, автор, название, год издания; сведения о читателе: номер читательского билета, фамилия; дата заказа);
5. Экзаменационная ведомость (предмет, номер группы, дата экзамена, 25 строчек с полями: фамилия студента, номер его зачетной книжки, оценка за экзамен);
6. Сведения о 10 студентах группы (фамилия, имя, отчество; дата рождения: число, месяц, год; пол; семейное положение; отношение к воинской службе);
7. Сведения о заводе-изготовителе компьютеров (название завода; название заводо-поставщиков комплектующих изделий; номера магазинов потребителей; название производимых компьютеров (5.. 10 наименований): цена каждого товара за штуку в рублях и в долларах; количество каждого вида изделий в штуках);
8. Сведения о 5 персонах (фамилия, имя, отчество, пол, национальность, семейное положение, образование (высшее, среднее специальное, среднее, неполное среднее); адрес: улица, номер дома, номер квартиры, почтовый индекс, номер телефона).
9. Сведения о заводе-изготовителе автомобилей (название завода, название заводо-поставщиков комплектующих изделий, номера фирменных магазинов, марка производимых автомобилей, цена автомобиля в рублях и в долларах, количество каждой выпускаемой марки автомобилей).
10. Сведения об успеваемости студентов (фамилия, имя, отчество, номер зачетной книжки, список оценок по предметам).

Тема 8. Связанные динамические структуры данных. Методы программирования с

использованием связанных динамических структур данных

1. По списку L постройте два новых списка: L1 – из положительных элементов и L2 – из остальных элементов списка L (тип элементов – Real).
2. Найдите среднее арифметическое всех элементов непустого списка L (тип элементов – Real).
3. Объедините два упорядоченных по неубыванию списка L1 и L2 (тип элементов – Real) в один новый упорядоченный по неубыванию список L.
4. Включите в упорядоченный список элемент, сохраняя упорядоченность.
5. Проверьте на равенство две очереди. Решение в программе оформите через подпрограммы.
6. Найдите все вершины заданного графа, недостижимые от заданной его вершины.
7. Для двух выделенных вершин графа постройте соединяющий их простой путь.
8. Задана система односторонних дорог. Найдите путь, соединяющий города A и B и не проходящий через заданное множество городов.
9. Задана система двусторонних дорог. Найдите два города и соединяющий их путь, который проходит через каждую из дорог системы ровно один раз.
10. Задана система двусторонних дорог, причем для любой пары городов можно указать соединяющий их путь. Найдите такой город, для которого сумма расстояний до остальных городов минимальна.

Тема 9. Понятие и методы объектно-ориентированного программирования

1. Student: Фамилия, Имя, Отчество, Дата рождения, Адрес, Телефон, Факультет, Курс.
Создать массив объектов. Вывести:
 - а) список студентов заданного факультета;
 - б) списки студентов для каждого факультета и курса;
 - в) список студентов, родившихся после заданного года.
2. Abiturient: Фамилия, Имя, Отчество, Адрес, Оценки. Создать массив объектов. Вывести:
 - а) список абитуриентов, имеющих неудовлетворительные оценки;
 - б) список абитуриентов, сумма баллов у которых не меньше заданной;
 - в) выбрать N абитуриентов, имеющих самую высокую сумму баллов, и список абитуриентов, имеющих полупроходной балл.
3. Aeroflot: Пункт назначения, Номер рейса, Тип самолета, Время вылета, Дни недели. Создать массив объектов. Вывести:

══

- а) список рейсов для заданного пункта назначения;
- б) список рейсов для заданного дня недели;
- в) список рейсов для заданного дня недели, время вылета для которых больше заданного.

4. Book: Автор, Название, Издательство, Год, Количество страниц.

Создать массив объектов. Вывести:

- а) список книг заданного автора;
- б) список книг, выпущенных заданным издательством;
- в) список книг, выпущенных после заданного года.

5. Worker: Фамилия и инициалы, Должность, Год поступления на работу, Зарплата. Создать массив объектов. Вывести:

- а) список работников, стаж работы которых на данном предприятии превышает заданное число лет;
- б) список работников, зарплата которых больше заданной;
- в) список работников, занимающих заданную должность.

6. Train: Пункт назначения, Номер поезда, Время отправления, Число общих мест, Купейных, Плацкартных. Создать массив объектов. Вывести:

- а) список поездов, следующих до заданного пункта назначения;
- б) список поездов, следующих до заданного пункта назначения и отправляющихся после заданного часа;
- в) список поездов, отправляющихся до заданного пункта назначения и имеющих общие места.

7. Product: Наименование, Производитель, Цена, Срок хранения, Количество. Создать массив объектов. Вывести:

- а) список товаров для заданного наименования;
- б) список товаров для заданного наименования, цена которых не превышает указанной;
- в) список товаров, срок хранения которых больше заданного.

8. Patient: Фамилия, Имя, Отчество, Адрес, Номер медицинской карты, Диагноз.

Создать массив объектов. Вывести:

- а) список пациентов, имеющих данный диагноз;
- б) список пациентов, номер медицинской карты которых находится в заданном интервале.

9. Bus: Фамилия и инициалы водителя, Номер автобуса, Номер маршрута, Марка, Год

==

начала эксплуатации, Пробег. Создать массив объектов. Вывести:

- а) список автобусов для заданного номера маршрута;
- б) список автобусов, которые эксплуатируются больше 10 лет;
- в) список автобусов, пробег у которых больше 10 000 км.

10. Customer: Фамилия, Имя, Отчество, Адрес, Телефон, Номер

кредитной карточки, Номер банковского счета. Создать массив объектов. Вывести:

- а) список покупателей в алфавитном порядке;
- б) список покупателей, номер кредитной карточки которых находится в заданном

интервале.

Тема 10. Операторный базис языка программирования C++

1. Вычислите длину окружности, площадь круга и объём шара одного и того же заданного радиуса.
2. Определите номер квадранта, в котором находится точка с заданными координатами (x, y) .
3. Подсчитайте число и сумму положительных, число и произведение отрицательных элементов заданного массива $A(N)$.
4. Запишите подряд в массив $A(N)$ элементы заданного массива $B(2N)$, стоящие на чётных местах, а элементы, стоящие на нечётных местах, запишите в массив $C(N)$.
5. Дана матрица $A(N, M)$. Найдите её наибольший элемент и номера строки и столбца, на пересечении которых он находится.
6. Дана матрица $A(N, N)$ и целое P . Преобразуйте матрицу по правилу: строку с номером P сделайте столбцом с номером P , а столбец с номером P сделайте строкой с номером P .
7. Выясните, имеются ли в заданном векторе $A(N)$ два подряд идущих нулевых элемента.
8. Все элементы заданного вектора $A(N)$, начиная с первого по порядку положительного элемента, уменьшите на единицу.
9. В заданной целочисленной матрице $A(N, M)$ выведите на печать индексы первого положительного элемента, кратного заданному числу K . Если таких элементов в матрице нет, то выведите соответствующий текст. Элементы матриц просматривайте слева направо и сверху вниз.
10. Элементы заданной матрицы $A(N, N)$ переписывайте построчно в одномерный массив до тех пор, пока не встретится нулевой элемент.
11. В заданной матрице $A(N, M)$ найдите количество строк, не содержащих отрицательных чисел.

==

12. Дана матрица $A (N, M)$. Постройте вектор $B(N)$, элементы B_i которого равны единице, если элементы i -й строки образуют упорядоченную по убыванию или по возрастанию последовательность, и нулю во всех остальных случаях.

Тема 11. Методы трансляции

1. Постройте синтаксический анализатор для понятия *простое-выражение*:

простое выражение ::=

$$\{ \text{простой-идентификатор} \mid \text{простое-выражение} \text{ знак-операции } \text{простое-выражение} \}$$

простой идентификатор ::= буква

$$\text{знак операции} ::= \{ \text{+} \mid \text{-} \mid \text{*} \mid \text{/} \}$$

2. Постройте синтаксический анализатор для понятия *идентификатор*:

$$\text{идентификатор} ::= \{ \text{буква} \mid \text{идентификатор} \text{ цифра} \mid \text{цифра} \}$$

3. Постройте синтаксический анализатор для понятия *вещественное число*:

вещественное число ::=

$$\{ \text{целое} \mid \text{целое} \text{ без-знака} \mid \text{целое} \text{ без-знака} \text{ точка} \text{ целое} \mid \text{без-знака} \text{ без-знака} \text{ точка} \text{ без-знака} \text{ целое} \mid \text{без-знака} \text{ без-знака} \text{ точка} \text{ без-знака} \text{ точка} \text{ целое} \mid \text{без-знака} \text{ без-знака} \text{ точка} \text{ без-знака} \text{ точка} \text{ точка} \text{ целое} \}$$

целое-без-знака ::= цифра {цифра}*

$$\text{целое-число} ::= \{ \text{целое} \mid \text{без-знака} \text{ целое} \mid \text{без-знака} \text{ без-знака} \}$$

4. Постройте синтаксический анализатор для понятия *константное выражение*:

константное-выражение ::=

$$\{ \text{цифра} \mid \text{буква} \mid \text{знак-операции} \}$$

5. Постройте синтаксический анализатор для понятия *простое-логическое*:

простое-логическое ::=

$$\{ \text{простой-идентификатор} \mid \text{простое-логическое} \text{ знак-операции} \text{ простое-логическое} \mid \text{простое-логическое} \text{ знак-операции} \text{ простое-логическое} \text{ знак-операции} \text{ простое-логическое} \}$$

простой-идентификатор ::= буква

$$\text{знак-операции} ::= \{ \text{AND} \mid \text{OR} \}$$

6. Арифметическое выражение $(a+b)/2/f+(e+w)+c+g$, записанное в инфиксной форме, при переводе в постфиксную польскую нотацию будет иметь вид ...

- $ab+2/f+e+wc+g++$
- $ab+2/f+e+w+c+g+/$
- $ab+2/f+ew+c+g++$
- $ab+2/f+(ew+)c+g+$

7. Арифметическое выражение $a b + c * 1 +$, записанное в постфиксной польской нотации, при переводе в инфиксную форму будет иметь вид ...

- $a+b*(c+1)$
- $(a+b)*c+1$
- $a+(b*c)+1$
- $a+b*c+1$

8. Арифметическое выражение $1+(c-a*b)/2+x/y$, записанное в инфиксной форме, при переводе в постфиксную польскую нотацию будет иметь вид ...

- $1cab*-2/+xy/+$
- $cab*-2/+xy/+1$
- $1cab*-2/+xy+1$
- $1cab*-2/(xy)+$

9. Арифметическое выражение $r/t/a b + * 1 -$, записанное в постфиксной польской нотации, при переводе в инфиксную форму будет иметь вид ...

- $r/t*a+b-1$
- $r/t*a+(b-1)$
- $r/t+a*b-1$
- $r/t*(a+b)-1$

10. Арифметическое выражение $((a+c/d+1)*f+1)*q$, записанное в инфиксной форме, при переводе в постфиксную польскую нотацию будет иметь вид ...

- $ac/d+1+f*+1q*$
- $acd/+1+f*1+q*$
- $ac/d+1+f*1+q*$
- $acd/+1+*f1+q*$

11. Задана ПОЛИЗ некоторого простого выражения, не содержащего вхождений идентификаторов. Вычислите значение этого выражения.

12. Преобразуйте в ПОЛИЗ заданное простое выражение.

13. Восстановите простое выражение по заданной его ПОЛИЗ.

14. Задана прямая польская запись некоторого простого выражения, не содержащего вхождений идентификаторов. Вычислите значение этого выражения.

15. Преобразуйте в прямую польскую запись заданное простое выражение.

16. Восстановите простое выражение по заданной его прямой польской записи.

Методические рекомендации по решению задач

При решении задач необходимо внимательно ознакомиться с контентом по соответствующему вопросу темы. Разобрать основные понятия темы, разобрать примеры из

лабораторных работ. Кратко описать алгоритм решения задачи, объяснить критерии выбора соответствующей структуры данных. Выполнить кодирование задачи на языке программирования. Выполнить отладку и тестирование программы. В отчет включить листинг программы, полный набор тестов и результаты работы программы на этих тестах.

Критерии формирования оценок по заданиям для самостоятельной работы обучающегося (типовые задачи)

Самостоятельная работа оценивается степенью освоения вопросов для самостоятельного изучения и индивидуальным выполнением заданий к практическим занятиям.

В результате знания обучающегося оцениваются по ниже следующей шкале

Количество баллов	Критерии оценивания
5	Обучающийся показал глубокие знания материала по поставленным вопросам, грамотно, логично его излагает, свободно использует необходимые формулы при решении задач.
4	Обучающийся твердо знает материал, грамотно его излагает, не допускает существенных неточностей в процессе решения задач;
3	Обучающийся имеет знания основного материала по поставленным вопросам, но не усвоил его деталей, допускает отдельные неточности при решении задач.
2	Обучающийся имеет неполное знание и понимание основного материала по поставленным вопросам, не усвоил его деталей, допускает неточности при решении задач.
1	Обучающийся обнаруживает значительное незнание и понимание основного материала по поставленным вопросам, не усвоил его деталей, допускает существенные неточности при решении задач.
0	Обучающийся допускает грубые ошибки в ответе на поставленные вопросы и при решении задач.

5.2. *Оценочные материалы для рубежного контроля*

Рубежный контроль осуществляется по более или менее самостоятельным разделам – учебным модулям курса и проводится по окончании изучения материала модуля в заранее установленное время. Рубежный контроль проводится с целью определения качества усвоения материала учебного модуля в целом. В течение семестра проводится *три таких контрольных мероприятия по графику*.

В качестве форм рубежного контроля можно использовать тестирование (письменное или компьютерное), проведение коллоквиума или контрольных работ. Выполняемые работы должны храниться на кафедре течении учебного года и по требованию предоставляться в Управление контроля качества. На рубежные контрольные мероприятия

рекомендуется выносить весь программный материал (все разделы) по дисциплине.

5.2.1. Оценочные материалы для контрольной работы, коллоквиума

(контролируемая компетенция ПКС-2)

Типовые варианты контрольных работ

Вариант 1.

Задание 1. Общая структура программы на языке Паскаль.

Задание 2. Вычислить длину окружности, площадь круга и объем шара заданного радиуса.

Задание 3. Проверить, можно ли построить прямоугольный треугольник из отрезков с длинами a , b , c .

Вариант 2.

Задание 1. Операторы цикла в языке Паскаль.

Задание 2. В заданном массиве $A(N)$ подсчитать сумму элементов, кратных 5.

Задание 3. В заданной матрице $A(N, M)$ подсчитать количество строк, состоящих только из четных элементов.

Вариант 3.

Задание 1. Процедуры и функции обработки строк.

Задание 2. В заданном тексте найти самое длинное симметричное слово.

Задание 3. Упорядочить слова заданного текста в алфавитном порядке.

Вариант 4.

Задание 1. Процедуры и функции обработки файлов.

Задание 2. Найти максимальный компонент заданного файла.

Задание 3. Объединить два заданных файла и записать в третий файл.

Вариант 5.

Задание 1. Структура модуля.

Задание 2. Разработать текст модуля для решения геометрических задач.

Задание 3. Составить программу решение задачи с использованием модуля.

Вариант 6.

Задание 1. Динамические структуры данных.

Задание 2. Создать стек для хранения натуральных чисел.

Задание 3. Создать дерево двоичного поиска из заданного списка вещественных чисел.

Вариант 7.

Задание 1. Рекурсия.

Задание 2. Составить рекурсивную и не рекурсивную функцию вычисления n -го элемента ряда Фибоначчи и сравнить их время работы для различных входных данных.

Задание 3. В заданном графе найти самый короткий путь между двумя вершинами.

Критерии формирования оценок по контрольным точкам (контрольные работы; коллоквиум)

(5 баллов) - ставится за работу, выполненную полностью без ошибок и недочетов; обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, решено 100% задач;

(4 балла) – ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов. Обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, допуская незначительные неточности при решении задач, решено 70% задач;

(3 балла) – ставится за работу, если бакалавр правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой. Обучающийся затрудняется с правильной оценкой предложенной задачи, дает неполный ответ, решено 55% задач

(менее 3 баллов) – ставится за работу, если число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы. Обучающийся дает неверную оценку ситуации, решено менее 50 % задач.

5.2.2. Оценочные материалы: Типовые тестовые задания по дисциплине

«Языки и методы программирования» (контролируемые компетенции ПКС-3)

I: 55

S: Пусть вещественные a , b и c имеют значения 3, 5 и 7, соответственно. В результате выполнения последовательности операторов $a:=a+1$; $b:=a+b$; $c:=a+b$; $a:=\text{sqrt}(a)$; $\text{WriteLn}(a:4:1, b:4:1, c:4:1)$ на экран будет выведено ...

-: 2 9 13

+: 2.0 9.013.0

-: 2.0 5.0 13.0

-: 3.0 5.0 7.0

I: 59

S: При исполнении фрагмента программы $\text{ReadLn}(X)$; $\text{If } X \bmod 2=0 \text{ then WriteLn('ДА')}$ $\text{else WriteLn('НЕТ')}$ ответ "ДА" будет получен при X , равном ...

-: 3

-: 7

-: 15

+: 4

I: 60

S: Синтаксически неправильной является запись ...

-: If a>b then If b>c then If c>d then WriteLn(x);

-: If a>b then begin If b>c then If c>d then WriteLn(x) end;

-: If a>b then If b>c then If c>d then WriteLn(x) else WriteLn(y);

+: If a>b then x:=0; else If b>c then WriteLn(x) else WriteLn(y);

I: 64

S: Программа Var K,M: Integer; Begin K:=3; M:=25; If M mod 6<>M mod 8 then

K:=K+1; Write(K) End. печатает:

-: 5

-: 0

-: 1

+: 3

I: 79

S: Последовательность команд, в результате выполнения которых значения переменных X и Y поменяются местами:

+: X:=X+Y; Y:=X-Y; X:=X-Y;

-: B:=X; X:=Y; Y:=X;

-: X:=Y; Y:=X;

-: Y:=X; B:=X; X:=Y;

-: C:=X; X:=Y; X:=C;

I: 81

S: При исполнении фрагмента программы For i:=1 to 3 do S:=0; For j:=2 to 3 do begin S:=S+i+j; Write(S, ' ') end; на экран будут выведены числа ...

+: 5 11

-: 7 9 11

-: 3 7 4 9 5 11

-: 7 9

I: 91

S: В программе вычисления 10! Var i:Integer; f:Longint; Begin ... For i:=1 to 10 do f:=f*i; Write(f) End. недостает оператора ...

==

-: readln(N);

-: f:=0;

+: f:=1;

-: f:=N;

-: Нет никакой пропущенной строки

I: 98

S: Элементы квадратной матрицы, заданные фрагментом программы For i:=1 to n do

For j:=1 to n do If i=j then <операторы>, ...

+: расположены на главной диагонали

-: расположены выше главной диагонали

-: расположены на побочной диагонали

-: расположены на главной диагонали и выше ее

-: расположены ниже главной диагонали

I: 111

S: "Бесконечно" будут работать циклы ...

-: for i:=1 to i+1 do k:=k+1

+: for i:='x' to 'z' do dec(i)

-: for i:='x' downto 'z' do dec(i)

-: for i:=1 to n do n:=n+1

I: 206

S: Процедура Val(S: String; Var X, Code: Integer) ...

-: возвращает представление числа X в его символьной форме S

-: возвращает символ с заданным порядковым номером X

-: возвращает количество символов в строке S

+: возвращает представление символов строки S в ее числовой форме X.

I: 344

S: Для создания файла данных используются шаги: 1) открыть новый файл; 2) связать имя файла с файловой переменной; 3) записать в файл компоненты; 4) подготовить информацию; 5) закрыть созданный файл. Правильна последовательность шагов ...

-: 1, 2, 3, 4, 5

+: 2, 1, 4, 3, 5

-: 1, 2, 3, 5, 4

-: 2, 1, 4, 3, 5

==

I: 555

S: Программа Type den=(pon, vt, sr, ch, pt, sb, vos); k=[^]den; Var i,j :den; l,m :k; p,q:Byte; Begin p:=ord(vt); q:=ord(pt); i:=vos; j:=sb; l:=@i; m:=@j; Writeln(odd(p)=odd(q), 'ord(l[^]),', ord(m[^])) End. напечатает:

+: FALSE 6 5

-: TRUE l m

-: TRUE m l

-: FALSE sb vos

I: 900

S: Если 9/10 программы выполняется параллельно, а 1/10 последовательно, то в принципе невозможно получить ускорения более, чем ...

-: в 9/10 раз

-: в 9 раз

-: в 1/10 раз

+: в 10 раз

Критерии формирования оценок по тестовым заданиям:

5 баллов – получают обучающиеся с правильным количеством ответов на тестовые вопросы. Выполнено 100 % предложенных тестовых вопросов;

4 балла – получают обучающиеся с правильным количеством ответов на тестовые вопросы – 80 –99 % от общего объема заданных тестовых вопросов;

3 балла – получают обучающиеся с правильным количеством ответов на тестовые вопросы – 60 –79% от общего объема заданных тестовых вопросов;

1-2 балла – получают обучающиеся правильным количеством ответов на тестовые вопросы – менее 40-59 % от общего объема заданных тестовых вопросов.

5.2.3. Оценочные материалы для промежуточной аттестации (контролируемая компетенция ПКС-3)

Целью промежуточных аттестаций по дисциплине является оценка качества освоения дисциплины обучающимися.

Промежуточная аттестация предназначена для объективного подтверждения и оценивания достигнутых результатов обучения после завершения изучения дисциплины. Осуществляется в конце семестра и представляет собой итоговую оценку знаний по дисциплине «Языки и методы программирования» в виде проведения экзамена.

Для получения экзамена в 1,2 семестрах студенту необходимо иметь не менее 61 балла. Для допуска к экзамену студент должен по итогам текущего и рубежного контроля

==

успеваемости набрать число баллов не менее 36. На экзамене он может повысить сумму баллов от 61 и выше (до 100), необходимых для получения экзамена.

Вопросы, выносимые на экзамен по дисциплине «Языки и методы программирования» (контролируемая компетенция ПКС-3)

1. Концепция типов данных. Классификация типов данных в языке Паскаль.
2. Целочисленные типы данных. Диапазоны значений. Разрешенные операции.
3. Вещественные типы данных. Диапазоны значений. Разрешенные операции.
4. Символьный тип данных. Логический тип данных.
5. Перечисляемый и интервальный типы данных.
6. Алфавит и лексика языка Паскаль.
7. Идентификаторы. Знаки операций. Разделители. Комментарии.
8. Общая структура программы на языке Паскаль.
9. Раздел описаний в структуре программы.
10. Синтаксис описания модулей, меток, констант, типов, переменных.
11. Ввод и вывод данных в языке Паскаль.
12. Арифметические операции в языке Паскаль.
13. Побитовые операции в языке Паскаль.
14. Стандартные арифметические функции в языке Паскаль.
15. Логические операции и выражения. Приоритет операций.
16. Условный оператор. Оператор выбора.
17. Составной оператор. Оператор безусловного перехода. Пустой оператор.
18. Циклы в языке Паскаль (с параметром, с предусловием, с постусловием)
19. Синтаксис описания процедур.
20. Синтаксис описания функций.
21. Механизм передачи параметров в процедурах и функциях.
22. Область видимости и время жизни переменной.
23. Рекурсия в алгоритмах.
24. Опережающее объявление подпрограмм. Параметры-процедуры и параметры-функции.
25. функции.
26. Синтаксис описания массивов. Массивы как параметры подпрограмм.
27. Символьный тип данных `char`. Константы типа, разрешенные операции, функции для работы с типом.
28. Строковый тип `string`. Описание, внутренне представление, разрешенные операции.
29. Процедуры и функции обработки строк.

30. Множества. Синтаксис описания и операции над множествами.
31. Записи. Записи с вариантами. Примеры.
32. Язык программирования C. Достоинства и недостатки.
33. Алфавит, идентификаторы и служебные слова в языке C.
34. Запись целых, вещественных и строковых констант в языке C.
35. Определение переменных и констант в языке C.
36. Типы данных языка C. Модификаторы знака и размера.
37. Унарные и бинарные операции в языке C.
38. Поразрядные, логические и сдвиговые операции в языке C.
39. Разделители в языке C (скобки, запятая, точка с запятой, двоеточие, многоточие)
40. Присваивание в языке C. Дополнительные операции присваивания.
41. Приведение типов и правила преобразования типов в языке C.
42. Текст программы и препроцессор.
43. Структура программы на языке C.
44. Форматированный вывод в языке C.
45. Форматированный ввод в языке C.
46. Ветвление в языке C (условные оператор, оператор выбора, условная операция).
47. Циклы в языке C (циклы типа для, пока, до).
48. Операторы управления выполнением программы (goto, break, continue, return)
49. Массивы в языке C.
50. Определение и описание функций в языке C.
51. Фактические и формальные параметры функций. Указатели в параметрах функций.
52. Директивы препроцессора.
53. Указатели и адресная арифметика
54. Указатели и массивы.
55. Функции для выделения и освобождения памяти.
56. Хранение, ввод и вывод символьных и строковых данных.
57. Функции работы со строками в языке C.
58. Математические функции в языке C.
59. Классы памяти в языке C.
60. Функция main() и ее параметры.
61. Структуры и объединения в языке C.
62. Битовые поля.
63. Файловый ввод и вывод в языке C.

64. Стандартные файлы потоков и функции работы с ними.
65. Поточковый ввод и вывод в языке C++.
66. Обработка исключительных ситуаций в C++.
67. Работа с памятью в языке C++ (операции new и delete).

Критерии формирования оценок по промежуточной аттестации (для экзамена в случае, если экзаменационный билет содержит два вопроса)

Семестр	Шкала оценивания (по итогам текущего и рубежного контроля)			
	Неудовлетворит. (36-60 баллов)	Удовлетворит. (61-80 баллов)	Хорошо (81-90 баллов)	Отлично (91-100 баллов)
1, 2	<p>Обучающийся имеет 36-60 баллов по итогам текущего и рубежного контроля, на экзамене не дал полного ответа ни на один вопрос.</p> <p>Обучающийся имеет 36-45 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ только на один вопрос</p>	<p>Обучающийся имеет 36-50 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос и частично (полностью) ответил на второй.</p> <p>Обучающийся имеет 46-60 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос или частично ответил на оба вопроса.</p> <p>Обучающийся имеет по итогам текущего и рубежного контроля 61-70 баллов на экзамене не дал полного ответа ни на один вопрос</p>	<p>Обучающийся имеет 51-60 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос и частично (полностью) ответил на второй.</p> <p>Обучающийся имеет 61 – 65 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос и частично ответил на второй.</p> <p>Обучающийся имеет 66-70 баллов по итогам текущего и рубежного контроля, на экзамене) дал полный ответ только на один вопрос.</p>	<p>Обучающийся имеет 61-70 баллов по итогам текущего и рубежного контроля, на экзамене дал полный ответ на один вопрос и частично (полностью) ответил на второй.</p>

6. Методические материалы, определяющие процедуры оценивания знаний,

умений, навыков и (или) опыта деятельности

Учебная работа по дисциплине состоит из контактной работы (практические и лабораторные занятия) и самостоятельной работы. Максимальная сумма (100 баллов), набираемая студентом по дисциплине, включает две составляющие:

первая составляющая – оценка регулярности, своевременности и качества выполнения студентом учебной работы по изучению дисциплины в течение периода изучения дисциплины (семестра, или нескольких семестров) (сумма – не более 70 баллов). Баллы, характеризующие успеваемость студента по дисциплине, набираются им в течение всего периода обучения за изучение отдельных тем и выполнение отдельных видов работ.

вторая составляющая – оценка знаний студента по результатам промежуточной аттестации (не более 30 –баллов).

Критерием оценки уровня сформированности компетенций в рамках учебной дисциплины «Языки и методы программирования» в 1-2 семестрах является экзамен.

Распределение баллов текущего и рубежного контроля

№ п/п	Вид контроля	Сумма баллов			
		Общая сумма в баллах	1-я точка	2-я точка	3-я точка
1.	Посещение занятий	10	3	3	4
2.	Текущий контроль:	до 30	до 10	до 10	до 10
	Выполнение самостоятельных заданий (решение задач)	0 -15	0 – 5	0 -5	0 - 5
3.	Рубежный контроль	до 30	до 10	до 10	до 10
	<i>тестирование</i>	0- 12	0- 4	0- 4.	0- 4.
	<i>коллоквиум</i>	0 - 18	0 - 6	0 -6	0 - 6
4.	Итого сумма текущего и рубежного контроля	до 70	до 23	до 23	до 24
В случае экзамена					
5.	Первый этап (базовый уровень) – оценка «удовлетворительно»	не менее 36	не менее 12	не менее 12	не менее 12
6.	Второй этап (продвинутый уровень) – оценка «хорошо»	менее 70	менее 23	менее 23	менее 24
7.	Третий этап (высокий уровень) - оценка «отлично»	не менее 70	не менее 23	не менее 23	не менее 24

Типовые задания, обеспечивающие формирование компетенций ОПК-4 представлены в таблице 7.

Таблица 7. Результаты освоения учебной дисциплины, подлежащие проверке

Результаты обучения (компетенции)	Индикаторы достижения компетенции	Освоенные показатели оценки результатов обучения	Виды оценочного материала, обеспечивающий формирование компетенций
<p>ПКС-3. Способен преподавать математику и информатику в средней школе, специальных учебных заведениях на основе полученного фундаментального образования и научного мировоззрения</p>	<p>ПКС-3.1. Способен использовать методические основы преподавания профессиональных дисциплин</p>	<p>ПКС-3.1. З-1. методические основы преподавания дисциплин математики и информатики. ПКС-3.1. У-1. Профессионально грамотно пользоваться организационно-методическим и учебно-методическим обеспечением образовательной программы соответствующего уровня. ПКС-3.1. В-1. психолого-педагогическими и методическими основами преподавания дисциплин математики и информатики.</p>	<p>Типовые оценочные материалы для устного опроса (п. 5.1.1); типовые оценочные материалы для контрольной работы (п. 5.2.1); типовые тестовые задания (п. 5.2.2); типовые оценочные материалы к экзамену (п. 5.2.3)</p>
	<p>ПКС-3.2. Способен планировать лекционные и семинарские занятия по программам профессионального обучения математике и информатике, с учетом уровня подготовки и психологию аудитории</p>	<p>ПКС 3.2. З-1. образовательный стандарт и программы среднего общего образования, среднего профессионального образования и дополнительные общеобразовательные и профессиональные программы соответствующего уровня ПКС 3.2. У-1. Применять теоретические и практические основы по дополнительным разделам элементарной математики при проведении методических и экспертных работ. ПКС-3.2. В-1. Способностью к планированию и осуществлению педагогической деятельности с учетом специфики предметной области в образовательных организациях.</p>	

7. Учебно-методическое обеспечение дисциплины (модуля)

7.1. Нормативно-законодательные акты

1. Приказ Минобрнауки России от 06.04.2021 № 245 "Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры" (Зарегистрировано в Минюсте России 13.08.2021 N 64644).
2. Федеральный государственный образовательный стандарт по образовательным программам ВО (ФГОС 3++) по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии (уровень бакалавриата). Приказ Министерства образования и науки Российской Федерации от 10 января 2018г. №9 (Зарегистрировано в министерстве юстиции Российской Федерации 06 февраля 2018г. № 49937);
3. Федеральный закон "Об образовании в Российской Федерации" от 29.12.2012 N 273-ФЗ http://www.consultant.ru/document/cons_doc_LAW_140174/
4. Программа «Цифровая экономика», утвержденная распоряжением Правительства Российской Федерации от 28 июля 2017 г. №1632-р.
5. Указ Президента Российской Федерации от 9 мая 2017 г. №203 «О Стратегии развития информационного общества в Российской Федерации на 2017–2030 годы».

7.2. Основная литература

1. Кауфман В.Ш. Языки программирования. Концепции и принципы [Электронный ресурс] / В.Ш. Кауфман. — Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 464 с. — 978-5-4488-0137-2 iprbookshop.ru/64055.
2. Непейвода Н.Н. Стили и методы программирования [Электронный ресурс] / Н.Н. Непейвода. — 2-е изд. — Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 295 с. — 5-9556-0023-X. iprbookshop.ru/73724.
3. Молдованова О.В. Языки программирования и методы трансляции [Электронный ресурс] : учебное пособие / О.В. Молдованова. — Электрон. текстовые данные. — Новосибирск: Сибирский государственный университет телекоммуникаций и информатики, 2012. — 134 с. — 2227-8397. iprbookshop.ru/54809.
4. Обучение программированию. Язык Pascal [Электронный ресурс] / Н.А. Тарануха [и др.]. — Электрон. текстовые данные. — М.: СОЛОН-ПРЕСС, 2009. — 384 с. — 978-5-91359-050-3 iprbookshop.ru/8633.
5. Александров Э.Э. Программирование на языке C в Microsoft Visual Studio 2010 [Электронный ресурс] / Э.Э. Александров, В.В. Афонин. — 2-е изд. — Электрон.

текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 570 с. — 2227-8397 iprbookshop.ru/73712

7.3. Дополнительная литература

1. Никлаус Вирт Алгоритмы и структуры данных [Электронный ресурс] / Вирт Никлаус. — Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 272 с. — 978-5-4488-0101-3. iprbookshop.ru/63821
2. Технологии программирования [Электронный ресурс]: учебное пособие / А.А. Смирнов, Д.В. Хрипков. — Электрон. текстовые данные. — М.: Евразийский открытый институт, 2011. — 191 с. — 978-5-374-00296-6 iprbookshop.ru/10900
3. Теория и реализация языков программирования [Электронный ресурс] / В.А. Серебряков [и др.]. — 2-е изд. — Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 372 с. — 2227-839 iprbookshop.ru/73731
4. Методы программирования [Электронный ресурс]: учебное пособие / Е.В. Ковалевская, Н.В. Комлева. — Электрон. текстовые данные. — М.: Евразийский открытый институт, 2011. — 320 с. — 978-5-374-00356-7 iprbookshop.ru/10784
5. Алексеев Е.Р. Free Pascal и Lazarus [Электронный ресурс]: учебное пособие по программированию / Е.Р. Алексеев, О.В. Чеснокова, Т.В. Кучер. — Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 438 с. — 978-5-4488-0105-1 iprbookshop.ru/63825
6. Бедняк С.Г. Решение задач на ЭВМ. Программирование на языке Pascal [Электронный ресурс]: учебное пособие / С.Г. Бедняк, О.И. Захарова. — Электрон. текстовые данные. — Самара: Поволжский государственный университет телекоммуникаций и информатики, 2014. — 198 с. — 978-5-904029-44-9. iprbookshop.ru/71875
7. Зюзьков В.М. Программирование [Электронный ресурс]: учебное пособие / В.М. Зюзьков. — Электрон. текстовые данные. — Томск: Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2013. — 186 с. — 978-5-4332-0141-5. iprbookshop.ru/72168
8. Разработка компиляторов [Электронный ресурс] / Н.Н. Вояковская [и др.]. — 2-е изд. — Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 374 с. — 2227-8397 iprbookshop.ru/73654
9. Федотов И.Е. Приемы параллельного программирования [Электронный ресурс]: учебное пособие / И.Е. Федотов. — Электрон. текстовые данные. — М.: Российский

новый университет, 2009. — 184 с. — 978-5-89789-048-4. iprbookshop.ru/21300

10. Малявко А.А. Формальные языки и компиляторы [Электронный ресурс]: учебник / А.А. Малявко. — Электрон. текстовые данные. — Новосибирск: Новосибирский государственный технический университет, 2014. — 431 с. — 978-5-7782-2318-9 iprbookshop.ru/47725
11. Кивран В.К. Программирование в среде Visual C++ 6 [Электронный ресурс]: учебное пособие / В.К. Кивран. — Электрон. текстовые данные. — Самара: Самарский государственный архитектурно-строительный университет, ЭБС АСВ, 2014. — 118 с. — 978-5-9585-0601-9 iprbookshop.ru/43185

7.4. Периодические издания

1. Журнал «Математическое моделирование»
2. Журнал «Информатика и управление»

7.5. Интернет-ресурсы

1. <http://www.cyberforum.ru/> - Форум программистов и сисадминов «Киберфорум»
2. <http://intuit.ru> – Национальный Открытый Университет «Интуит»
3. <http://www.osp.ru> – Издательство «Открытые системы»
4. <http://www.cnews.ru> – Издание о высоких технологиях
5. <http://habrahabr.ru/> - Сообщество IT-профессионалов

При проведении занятий лекционного типа практических (семинарских) занятий используются сведения об электронных информационных ресурсах, к которым обеспечен доступ для пользователей библиотеки КБГУ.

Перечень актуальных электронных информационных баз данных, к которым обеспечен доступ пользователям КБГУ

(2022-2023 уч. год)

№ п/п	Наименование электронного ресурса	Краткая характеристика	Адрес сайта	Наименование организации-владельца; реквизиты договора	Условия доступа
--------------	--	-------------------------------	--------------------	---	------------------------

1.	Научная электронная библиотека (НЭБ РФФИ)	Электр. библиотека научных публикаций - около 4000 иностранных и 3900 отечественных научных журналов, рефераты публикаций 20 тыс. журналов, а также описания 1,5 млн. зарубежных и российских диссертаций; 2800 росс. журналов на безвозмездной основе	http://elibrary.ru	ООО «НЭБ» Лицензионное соглашение №14830 от 01.08.2014. Бессрочное	Полный доступ
1.	База данных Science Index (РИНЦ)	Национальная информационно-аналитическая система, аккумулирующая более 6 миллионов публикаций российских авторов, а также информацию об их цитировании из более 4500 российских журналов.	http://elibrary.ru	ООО «НЭБ» Лицензионный договор Science Index №SIO-741/2022 от 19.07.2022 Активен до 31.07.2023	Авторизованный доступ. Позволяет дополнять и уточнять сведения о публикациях ученых КБГУ, имеющихся в РИНЦ
2.	ЭБС «Консультант студента»	13800 изданий по всем областям знаний, включает более чем 12000 учебников и учебных пособий для ВО и СПО, 864 наименований журналов и 917 монографий.	http://www.studmedlib.ru http://www.medcollegelib.ru	ООО «Консультант студента» (г. Москва) Договор №750КС/07-2022 От 26.09.2022 Активен до 30.09.2023	Полный доступ (регистрация по IP-адресам КБГУ)

3.	«Электронная библиотека технического вуза» (ЭБС «Консультант студента»)	Коллекция «Медицина (ВО) ГЭОТАР-Медиа. Books in English (книги на английском языке)»	http://www.studmedlib.ru	ООО «Политехресурс» (г. Москва) Договор №849КС/03-2023 от 11.04.2023 Активен до 19.04.2024	Полный доступ (регистрация по IP-адресам КБГУ)
4.	ЭБС «Лань»	Электронные версии книг ведущих издательств учебной и научной литературы (в том числе университетских издательств), так и электронные версии периодических изданий по различным областям знаний.	https://e.lanbook.com/	ООО «ЭБС ЛАНЬ» (г. Санкт-Петербург) Договор №41ЕП/223 от 14.02.2023 Активен до 15.02.2024	Полный доступ (регистрация по IP-адресам КБГУ)
5.	Национальная электронная библиотека РГБ	Объединенный электронный каталог фондов российских библиотек, содержащий 4 331 542 электронных документов образовательного и научного характера по различным отраслям знаний	https://rusneb.ru/	ФГБУ «Российская государственная библиотека» Договор №101/НЭБ/166 6-п от 10.09.2020 Бессрочный	Доступ с электронного читального зала библиотеки КБГУ
6.	ЭБС «IPSMART»	107831 публикаций, в т.ч.: 19071 – учебных изданий, 6746 – научных изданий, 700 коллекций, 343 журнала ВАК, 2085 аудиозданий.	http://iprbookshop.ru/	ООО «Ай Пи Эр Медиа» (г. Москва) Договор №75/ЕП-223 от 23.03.2023 Активен до 02.04.2024	Полный доступ (регистрация по IP-адресам КБГУ)

7.	ЭБС «Юрайт» для СПО	Электронные версии учебной и научной литературы издательств «Юрайт» для СПО и электронные версии периодических изданий по различным областям знаний.	https://urait.ru/	ООО «Электронное издательство ЮРАЙТ» (г. Москва) Договор №305/ЕП-223 От 27.10.2022 Активен до 31.10.2023.	Полный доступ (регистрация по IP-адресам КБГУ)
8.	ЭБС «Юрайт» для ВО	Электронные версии 8000 наименований учебной и научной литературы издательств «Юрайт» для ВО и электронные версии периодических изданий по различным областям знаний.	https://urait.ru/	ООО «Электронное издательство ЮРАЙТ» (г. Москва) Договор №44/ЕП-223 От 16.02.2023 Активен с 01.03.2023 по 29.02.2024	Полный доступ (регистрация по IP-адресам КБГУ)
9.	Polpred.com. Новости. Обзор СМИ. Россия и зарубежье	Обзор СМИ России и зарубежья. Полные тексты + аналитика из 600 изданий по 53 отраслям	http://polpred.com	ООО «Полпред справочники» Безвозмездно (без официального договора)	Доступ по IP-адресам КБГУ
10.	Президентская библиотека им. Б.Н. Ельцина	Более 500 000 электронных документов по истории Отечества, российской государственности, русскому языку и праву	http://www.prilib.ru	ФГБУ «Президентская библиотека им. Б.Н. Ельцина» (г. Санкт-Петербург) Соглашение от 15.11.2016г. Бессрочный	Авторизованный доступ из библиотеки (ауд. №115, 214)

7.6. Методические указания по проведению различных учебных занятий, к курсовому проектированию и другим видам самостоятельной работы

Учебная работа по дисциплине состоит из контактной работы (практические и лабораторные занятия) и самостоятельной работы.

Для подготовки к лабораторным занятиям необходимо рассмотреть контрольные

вопросы, при необходимости обратиться к рекомендуемой литературе, записать непонятные моменты в вопросах для уяснения их на предстоящем занятии.

Методические рекомендации по изучению дисциплины «Языки и методы программирования» для обучающихся

Цель курса «Языки и методы программирования» - ознакомление студентов с основными понятиями, структурами, методами, алгоритмами, парадигмами современного программирования и приобретение навыков решения задач с использованием современных языков и технологий программирования.

Приступая к изучению дисциплины, обучающемуся необходимо ознакомиться с тематическим планом занятий, списком рекомендованной учебной литературы. Следует уяснить последовательность выполнения индивидуальных учебных заданий, занести в свою рабочую тетрадь темы и сроки проведения семинаров, написания учебных и творческих работ. При изучении дисциплины обучающиеся выполняют следующие задания: изучают рекомендованную учебную и научную литературу; пишут контрольные работы, готовят доклады и сообщения к практическим занятиям; выполняют самостоятельные творческие работы, участвуют в выполнении практических заданий. Уровень и глубина усвоения дисциплины зависят от активной и систематической работы на лекциях, изучения рекомендованной литературы, выполнения контрольных письменных заданий

Курс изучается на семинарах, при самостоятельной и индивидуальной работе обучающихся. Обучающийся для полного освоения материала должен не пропускать занятия и активно участвовать в учебном процессе. Лекции включают все темы и основные вопросы теории и практики страхования. Для максимальной эффективности изучения необходимо постоянно вести конспект лекций, знать рекомендуемую преподавателем литературу, позволяющую дополнить знания и лучше подготовиться к семинарским занятиям.

В соответствии с учебным планом на каждую тему выделено необходимое количество часов практических занятий, которые проводятся в соответствии с вопросами, рекомендованными к изучению по определенным темам. Обучающиеся должны регулярно готовиться к семинарским занятиям и участвовать в обсуждении вопросов. При подготовке к занятиям следует руководствоваться конспектом лекций и рекомендованной литературой. Тематический план дисциплины, учебно-методические материалы, а также список рекомендованной литературы приведены в рабочей программе

В ходе изучения дисциплины обучающийся имеет возможность подготовить

реферат по выбранной из предложенного в Рабочей программе списка теме. Выступление с докладом по реферату в группе проводится в форме презентации с использованием мультимедийной техники.

Методические рекомендации при работе над конспектом во время проведения лекции

В процессе лекционных занятий целесообразно конспектировать учебный материал. Для этого используются общие и утвердившиеся в практике правила, и приемы конспектирования лекций:

Конспектирование лекций ведется в специально отведенной для этого тетради, каждый лист которой должен иметь поля, на которых делаются пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

Целесообразно записывать тему и план лекций, рекомендуемую литературу к теме. Записи разделов лекции должны иметь заголовки, подзаголовки, красные строки. Для выделения разделов, выводов, определений, основных идей можно использовать цветные карандаши и фломастеры.

Названные в лекции ссылки на первоисточники надо пометить на полях, чтобы при самостоятельной работе найти и вписать их. В конспекте дословно записываются определения понятий, категорий и законов. Остальное должно быть записано своими словами.

Каждому студенту необходимо выработать и использовать допустимые сокращения наиболее распространенных терминов и понятий.

Методические рекомендации по подготовке к практическим занятиям

Практические (семинарские) занятия – составная часть учебного процесса, групповая форма занятий при активном участии студентов. Практические занятия способствуют углубленному изучению наиболее сложных проблем науки и служит основной формой подведения итогов самостоятельной работы обучающихся. Целью практических занятий является углубление и закрепление теоретических знаний, полученных обучающимися на лекциях и в процессе самостоятельного изучения учебного материала, а, следовательно, формирование у них определенных умений и навыков.

В ходе подготовки к семинарскому занятию необходимо прочитать конспект лекции, изучить основную литературу, ознакомиться с дополнительной литературой, выполнить выданные преподавателем практические задания. При этом учесть рекомендации преподавателя и требования программы. Дорабатывать свой конспект лекции, делая в нем соответствующие записи из литературы.

Желательно при подготовке к практическим занятиям по дисциплине одновременно использовать несколько источников, раскрывающих заданные вопросы.

На практических занятиях обучающиеся учатся грамотно излагать проблемы, свободно высказывать свои мысли и суждения, рассматривают ситуации, способствующие развитию профессиональной компетентности. Следует иметь в виду, что подготовка к практическому занятию зависит от формы, места проведения семинара, конкретных заданий и поручений. Это может быть написание доклада, эссе, реферата (с последующим их обсуждением), коллоквиум.

Методические рекомендации по подготовке к лабораторным занятиям

Лабораторные занятия – составная часть учебного процесса, групповая форма занятий при активном участии студентов. Лабораторные занятия способствуют углубленному изучению наиболее сложных проблем науки и служат основной формой подведения итогов самостоятельной работы обучающихся. Целью лабораторных занятий является углубление и закрепление теоретических знаний, полученных обучающимися на лекциях и в процессе самостоятельного изучения учебного материала, а, следовательно, формирование у них

Желательно при подготовке к лабораторным занятиям по дисциплине одновременно использовать несколько источников, раскрывающих заданные вопросы.

На лабораторных занятиях обучающиеся учатся грамотно излагать проблемы, свободно высказывать свои мысли и суждения, рассматривают ситуации, способствующие развитию профессиональной компетентности.

Методические рекомендации по организации самостоятельной работы

Самостоятельная работа (по В.И. Далу «самостоятельный – человек, имеющий свои твердые убеждения») осуществляется при всех формах обучения: очной и заочной.

Самостоятельная работа обучающихся - способ активного, целенаправленного приобретения студентом новых для него знаний и умений без непосредственного участия в этом процесса преподавателей. Повышение роли самостоятельной работы обучающихся при проведении различных видов учебных занятий предполагает:

- оптимизацию методов обучения, внедрение в учебный процесс новых технологий обучения, повышающих производительность труда преподавателя, активное использование информационных технологий, позволяющих обучающемуся в удобное для него время осваивать учебный материал;

- широкое внедрение компьютеризированного тестирования;

- совершенствование методики проведения практик и научно-исследовательской

работы обучающихся, поскольку именно эти виды учебной работы в первую очередь готовят обучающихся к самостоятельному выполнению профессиональных задач;

– модернизацию системы курсового и дипломного проектирования, которая должна повышать роль студента в подборе материала, поиске путей решения задач.

Самостоятельная работа приводит студента к получению нового знания, упорядочению и углублению имеющихся знаний, формированию у него профессиональных навыков и умений. Самостоятельная работа выполняет ряд функций:

- развивающую;
- информационно-обучающую;
- ориентирующую и стимулирующую;
- воспитывающую;
- исследовательскую.

В рамках курса выполняются следующие виды самостоятельной работы:

1. Проработка учебного материала (по конспектам, учебной и научной литературе);
2. Выполнение разноуровневых задач и заданий;
3. Работа с тестами и вопросами для самопроверки;
4. Выполнение итоговой контрольной работы.

Студентам рекомендуется с самого начала освоения курса работать с литературой и предлагаемыми заданиями в форме подготовки к очередному аудиторному занятию. При этом актуализируются имеющиеся знания, а также создается база для усвоения нового материала, возникают вопросы, ответы на которые студент получает в аудитории.

Необходимо отметить, что некоторые задания для самостоятельной работы по курсу имеют определенную специфику. При освоении курса студент может пользоваться библиотекой вуза, которая в полной мере обеспечена соответствующей литературой. Значительную помощь в подготовке к очередному занятию может оказать имеющийся в учебно-методическом комплексе краткий конспект лекций. Он же может использоваться и для закрепления полученного в аудитории материала. Самостоятельная работа студентов предусмотрена учебным планом и выполняется в обязательном порядке. Задания предложены по каждой изучаемой теме и могут готовиться индивидуально или в группе. По необходимости студент может обращаться за консультацией к преподавателю. Выполнение заданий контролируется и оценивается преподавателем.

Для успешного самостоятельного изучения материала сегодня используются различные средства обучения, среди которых особое место занимают информационные

технологии разного уровня и направленности: электронные учебники и курсы лекций, базы тестовых заданий и задач. Электронный учебник представляет собой программное средство, позволяющее представить для изучения теоретический материал, организовать апробирование, тренаж и самостоятельную творческую работу, помогающее студентам и преподавателю оценить уровень знаний в определенной тематике, а также содержащее необходимую справочную информацию. Электронный учебник может интегрировать в себе возможности различных педагогических программных средств: обучающих программ, справочников, учебных баз данных, тренажеров, контролирующих программ.

Для успешной организации самостоятельной работы все активнее применяются разнообразные образовательные ресурсы в сети Интернет: системы тестирования по различным областям, виртуальные лекции, лаборатории, при этом пользователю достаточно иметь компьютер и подключение к Интернету для того, чтобы связаться с преподавателем, решать вычислительные задачи и получать знания. Использование сетей усиливает роль самостоятельной работы студента и позволяет кардинальным образом изменить методику преподавания.

Студент может получать все задания и методические указания через сервер, что дает ему возможность привести в соответствие личные возможности с необходимыми для выполнения работ трудозатратами. Студент имеет возможность выполнять работу дома или в аудитории. Большое воспитательное и образовательное значение в самостоятельном учебном труде студента имеет самоконтроль. Самоконтроль возбуждает и поддерживает внимание и интерес, повышает активность памяти и мышления, позволяет студенту своевременно обнаружить и устранить допущенные ошибки и недостатки, объективно определить уровень своих знаний, практических умений. Самое доступное и простое средство самоконтроля с применением информационно-коммуникационных технологий - это ряд тестов «on-line», которые позволяют в режиме реального времени определить свой уровень владения предметным материалом, выявить свои ошибки и получить рекомендации по самосовершенствованию.

Методические рекомендации по работе с литературой

Всю литературу можно разделить на учебники и учебные пособия, оригинальные научные монографические источники, научные публикации в периодической печати. Из них можно выделить литературу основную (рекомендуемую), дополнительную и литературу для углубленного изучения дисциплины.

Изучение дисциплины следует начинать с учебника, поскольку учебник – это книга, в которой изложены основы научных знаний по определенному предмету в

соответствии с целями и задачами обучения, установленными программой.

При работе с литературой необходимо учитывать, что имеются различные виды чтения, и каждый из них используется на определенных этапах освоения материала.

Предварительное чтение направлено на выявление в тексте незнакомых терминов и поиск их значения в справочной литературе. В частности, при чтении указанной литературы необходимо подробнейшим образом анализировать понятия.

Сквозное чтение предполагает прочтение материала от начала до конца. Сквозное чтение литературы из приведенного списка дает возможность студенту сформировать свод основных понятий из изучаемой области и свободно владеть ими.

Выборочное – наоборот, имеет целью поиск и отбор материала. В рамках данного курса выборочное чтение, как способ освоения содержания курса, должно использоваться при подготовке к практическим занятиям по соответствующим разделам.

Аналитическое чтение – это критический разбор текста с последующим его конспектированием. Освоение указанных понятий будет наиболее эффективным в том случае, если при чтении текстов студент будет задавать к этим текстам вопросы. Часть из этих вопросов сформулирована в ФОС в перечне вопросов для собеседования. Перечень этих вопросов ограничен, поэтому важно не только содержание вопросов, но сам принцип освоения литературы с помощью вопросов к текстам.

Целью *изучающего* чтения является глубокое и всестороннее понимание учебной информации. Есть несколько приемов изучающего чтения:

1. Чтение по алгоритму предполагает разбиение информации на блоки: название; автор; источник; основная идея текста; фактический материал; анализ текста путем сопоставления имеющихся точек зрения по рассматриваемым вопросам; новизна.

2. Прием постановки вопросов к тексту имеет следующий алгоритм:

- медленно прочитать текст, стараясь понять смысл изложенного;
- выделить ключевые слова в тексте;
- постараться понять основные идеи, подтекст и общий замысел автора.

3. Прием тезирования заключается в формулировании тезисов в виде положений, утверждений, выводов.

К этому можно добавить и иные приемы: прием реферирования, прием комментирования.

Важной составляющей любого солидного научного издания является список литературы, на которую ссылается автор. При возникновении интереса к какой-то обсуждаемой в тексте проблеме всегда есть возможность обратиться к списку

относящейся к ней литературы. В этом случае вся проблема как бы разбивается на составляющие части, каждая из которых может изучаться отдельно от других. При этом важно не терять из вида общий контекст и не погружаться чрезмерно в детали, потому что таким образом можно не увидеть главного.

Подготовка к экзамену должна проводиться на основе лекционного материала, материала практических занятий с обязательным обращением к основным учебникам по курсу. Это позволит исключить ошибки в понимании материала, облегчит его осмысление, прокомментирует материал многочисленными примерами.

Методические рекомендации для подготовки к экзамену

Экзамен в 1,2 семестрах является формой итогового контроля знаний и умений обучающихся по данной дисциплине, полученных на лекциях, практических занятиях и в процессе самостоятельной работы. Основой для определения оценки служит уровень усвоения обучающимися материала, предусмотренного данной рабочей программой. К экзамену допускаются студенты, набравшие 36 и более баллов по итогам текущего и промежуточного контроля. На экзамене студент может набрать от 15 до 30 баллов.

В период подготовки к экзамену обучающиеся вновь обращаются к учебно-методическому материалу и закрепляют промежуточные знания.

Подготовка обучающегося к экзамену включает три этапа:

- самостоятельная работа в течение семестра;
- непосредственная подготовка в дни, предшествующие экзамену по темам курса;
- подготовка к ответу на экзаменационные вопросы.

При подготовке к экзамену обучающимся целесообразно использовать материалы лекций, учебно-методические комплексы, нормативные документы, основную и дополнительную литературу.

На экзамен выносятся материал в объеме, предусмотренном рабочей программой учебной дисциплины за семестр. Экзамен проводится в письменной / устной форме.

При проведении экзамена в письменной (устной) форме, ведущий преподаватель составляет экзаменационные билеты, которые включают в себя: тестовые задания; теоретические задания; задачи или ситуации. Формулировка теоретических задания совпадает с формулировкой перечня экзаменационных вопросов, доведенных до сведения обучающихся накануне экзаменационной сессии. Содержание вопросов одного билета относится к различным разделам программы с тем, чтобы более полно охватить материал учебной дисциплины.

В аудитории, где проводится устный экзамен, должно одновременно находиться не более шести студентов на одного преподавателя, принимающего экзамен. На подготовку ответа на билет на экзамене отводится 40 минут.

При проведении письменного экзамена на работу отводится 60 минут.

Результат устного (письменного) экзамена выражается оценками:

Оценка «отлично» – от 91 до 100 баллов – теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы. Все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному. На экзамене студент демонстрирует глубокие знания предусмотренного программой материала, умеет четко, лаконично и логически последовательно отвечать на поставленные вопросы.

Оценка «хорошо» – от 81 до 90 баллов – теоретическое содержание курса освоено, необходимые практические навыки работы сформированы, выполненные учебные задания содержат незначительные ошибки. На экзамене студент демонстрирует твердые знания основного (программного) материала, умеет четко, грамотно, без существенных неточностей отвечать на поставленные вопросы.

Оценка «удовлетворительно» – от 61 до 80 баллов – теоретическое содержание курса освоено не полностью, необходимые практические навыки работы сформированы частично, выполненные учебные задания содержат грубые ошибки. На экзамене студент демонстрирует знание только основного материала, ответы содержат неточности, слабо аргументированы, нарушена последовательность изложения материала.

Оценка «неудовлетворительно» – от 36 до 60 баллов – теоретическое содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к существенному повышению качества выполнения учебных заданий. На экзамене студент демонстрирует незнание значительной части программного материала, существенные ошибки в ответах на вопросы, неумение ориентироваться в материале, незнание основных понятий дисциплины.

8. Материально-техническое обеспечение дисциплины

8.1. Требования к материально-техническому обеспечению

Для реализации рабочей программы дисциплины имеются учебные аудитории для проведения занятий, оснащенные оборудованием и техническими средствами обучения,

==

комплект учебной мебели (преподавательские стол, стул; столы и стулья для обучающихся), наборами демонстрационного оборудования и учебно-наглядных пособий по изучаемым разделам, обеспечивающие тематические иллюстрации по дисциплине «Языки и методы программирования».

Помещения для самостоятельной работы, оснащенные компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду КБГУ. Компьютерная техника обеспечена необходимым комплектом лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства.

Электронно-библиотечные системы и электронная информационно-образовательная среда КБГУ обеспечивают доступ (удаленный доступ) обучающимся, к современным профессиональным базам данных и информационным справочным системам.

Компьютерные классы

Оснащены оборудованием и техническими средствами обучения, комплектом учебной мебели, компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду КБГУ.

Компьютерная техника обеспечена необходимым комплектом лицензионного и свободно распространяемого программного обеспечения, (проектор, рулонный экран, меловая доска).

Лицензионное и свободно распространяемое программное обеспечение

№	Наименование программы, право использования которой предоставляется	Страна происхождения	Срок действия программного обеспечения
	Техническая поддержка для операционной системы Конфигурация: «Рабочая станция» <i>Сертификат на техническую поддержку операционной системы РЕД ОС. Конфигурация: «Рабочая станция»</i>	Российская Федерация	12 месяцев

2	<p>Лицензия на программное обеспечение для поиска заимствований в текстовых документах распространённых форматов</p> <p><i>Программная система для обнаружения текстовых заимствований в учебных и научных работах «Антиплагиат. Вуз 4.0»,</i></p> <p><i>Модуль поиска текстовых заимствований “Объединенная коллекция 2020»</i></p>	Российская Федерация	12 месяцев / по истечении 2000 проверок
3	<p>Система оптического распознавания текста</p> <p><i>SETERE OCR для РЭД ОС</i></p>	Российская Федерация	12 месяцев
4	<p>Редактор изображений</p> <p><i>AliveColors Business (лицензия для образовательных учреждений)</i></p>	Российская Федерация	бессрочные
5	<p>Лицензия на программное обеспечение средств антивирусной защиты</p> <p><i>Kaspersky Endpoint Security для бизнеса – Стандартный Russian Edition</i></p>	Российская Федерация	12 месяцев
6	<p>Пакет офисного программного обеспечения</p> <p><i>Р7-Офис.Профессиональный (Десктопная версия)</i></p>	Российская Федерация	12 месяцев
7	<p>Право использования программного обеспечения для планирования и проведения онлайн-мероприятий (трансляций, телемостов/ аудио-видеоконференций, вебинаров)</p> <p><i>Webinar Enterprise TOTAL 150 участников</i></p>	Российская Федерация	12 месяцев
8	<p>Пакет программного обеспечения для обработки растровых изображений</p> <p><i>Creative Cloud for enterprise All Apps ALL Multiple Platforms Multi European Languages Enterprise Licensing Subscription Renewal, право на использование-50 шт.</i></p> <p><i>Acrobat Pro DC for teams ALL Multiple Platforms Multi European Languages Team Licensing Subscription Renewal, право на использование – 5 шт.</i></p>	Соединенные Штаты Америки	12 месяцев

9.	Лицензия на программное обеспечение для векторного графического редактора для создания и редактирования графических схем, чертежей и блок-схем <i>Асмо-графический редактор</i>	Российская Федерация	бессрочные
10.	Предоставление неисключительных прав на использование программного обеспечения Системы <i>Spider Project Professional</i>	Российская Федерация	бессрочные

8.2. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Для обучающихся с ограниченными возможностями здоровья созданы специальные условия для получения образования. В целях доступности получения высшего образования по образовательным программам инвалидами и лицами с ограниченными возможностями здоровья университетом обеспечивается:

1. Альтернативная версия официального сайта в сети «Интернет» для слабовидящих;

2. Для инвалидов с нарушениями зрения (слабовидящие, слепые)

- присутствие ассистента, оказывающего обучающемуся необходимую помощь, дублирование вслух справочной информации о расписании учебных занятий; наличие средств для усиления остаточного зрения, брайлевской компьютерной техники, видеоувеличителей, программ невизуального доступа к информации, программ-синтезаторов речи и других технических средств приема-передачи учебной информации в доступных формах для обучающихся с нарушениями зрения;

- задания для выполнения на экзамене зачитываются ассистентом;

- письменные задания выполняются на бумаге, надиктовываются ассистенту обучающимся;

3. Для инвалидов и лиц с ограниченными возможностями здоровья по слуху (слабослышащие, глухие):

- на зачете/экзамене присутствует ассистент, оказывающий обучающемуся необходимую техническую помощь с учетом индивидуальных особенностей (он помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе

записывая под диктовку);

- зачет/экзамен проводится в письменной форме;

4. Для инвалидов и лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, созданы материально-технические условия, обеспечивающие возможность беспрепятственного доступа обучающихся в учебные помещения, объекты питания, туалетные и другие помещения университета, а также пребывания в указанных помещениях (наличие расширенных дверных проемов, поручней и других приспособлений).

- письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;

- по желанию обучающегося экзамен проводится в устной форме.

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

9. Лист изменений (дополнений)

в рабочей программе дисциплины «Языки и методы программирования» направления подготовки 02.03.02 Фундаментальная информатика и информационные технологии, профиль «Проектирование систем искусственного интеллекта» на 2023-2024 учебный год.

№ п/п	Элемент (пункт) РПД	Перечень вносимых изменений (дополнений)	Примечание
1.			
2.			
3.			

Обсуждена и рекомендована на заседании кафедры

Прикладной математики и информатики

Протокол №__ от «_____» _____ 2023г.

Зав. кафедрой _____ А.Р. Бечелова