

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования «Кабардино-Балкарский государственный университет
им. Х.М. Бербекова» (КБГУ)

ИНСТИТУТ ИНФОРМАТИКИ, ЭЛЕКТРОНИКИ И РОБОТОТЕХНИКИ
КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ИНФОРМАЦИОННОЙ
БЕЗОПАСНОСТИ

СОГЛАСОВАНО

Руководитель ОПОП

_____ Р.В.Гурфова

«____» _____ 2021 г.

УТВЕРЖДАЮ

Директор ИИЭР

_____ Н.В.
Черкесова

«____» _____ 2021 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ»

Направление подготовки (специальность)

09.03.03 – Прикладная информатика

Профиль подготовки:

«Корпоративные информационные системы»

Квалификация (степень) выпускника

Бакалавр

Форма обучения

очная

Нальчик 2021

Рабочая программа дисциплины «Практикум по программированию» /сост. Д.А. Крымшохалова – Нальчик: ФГБОУ КБГУ, 2021. – 30 с.

Рабочая программа предназначена для преподавания дисциплины обязательной части студентам очной формы обучения по направлению подготовки 09.03.03 Прикладная информатика, профиль Корпоративные информационные системы во 2 семестре 1 курса.

Рабочая программа составлена с учетом Федерального государственного образовательного стандарта высшего образования по направлению 09.03.03 Прикладная информатика (уровень бакалавриата), утвержденного приказом Министерства образования и науки Российской Федерации от «19» сентября 2017 г. № 922 (зарегистрировано в Минюсте России 12 октября 2017 г. № 48531)

СОДЕРЖАНИЕ

1. Цель и задачи освоения дисциплины.....	4
2. Место дисциплины в структуре ОПОП ВО.....	4
3. Требования к результатам освоения дисциплины.....	4
4. Содержание и структура дисциплины.....	6
5. Оценочные материалы для текущего и рубежного контроля успеваемости и промежуточной аттестации.....	10
6. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.....	20
7. Учебно-методическое обеспечение дисциплины.....	21
8. Материально-техническое обеспечение дисциплины.....	26
9. Лист изменений (дополнений).....	28
Приложение.....	29

1. Цель и задачи освоения дисциплины

Цель:

- ознакомить студентов с основными понятиями, структурами, методами и алгоритмами современного программирования;
- ознакомить студентов с различными парадигмами программирования; практически ознакомить студентов с языками программирования высокого уровня C++;
- ознакомить 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 объемом до нескольких сотен операторов с использованием современных технологий программирования.

Задачи:

- 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 и структур данных и их реализации на современных программных средствах.
- создать фундамент освоения новых языков программирования;
- способствовать развитию точного научного мышления, повышению программистской культуры.

Изучение дисциплины направлено на подготовку специалистов, способных решать проблемы и задачи профессиональной деятельности в соответствии с профессиональными стандартами:

- 06.015 «Специалист по информационным системам», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 18 ноября 2014 года N 896н (зарегистрирован в Минюсте РФ 24 декабря 2014 года, регистрационный N 35361).
- 06.016 «Руководитель проектов в области информационных технологий», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 18 ноября 2014 года N 893н (зарегистрирован Минюстом России 09.12.2014 г. № 35117).

2. Место дисциплины в структуре ОПОП ВО

Дисциплина относится к обязательной части учебного плана по направлению 09.03.01 Информатика и вычислительная техника, профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем» – Б1.О.05.07. Дисциплина является частью модуля «Информатика и программирование».

3. Требования к результатам освоения дисциплины

В совокупности с другими дисциплинами профиля «Корпоративные информационные системы» дисциплина «Практикум по программированию» направлена на формирование следующих компетенций в соответствии с ФГОС ВО и ОПОП ВО по направлению подготовки 09.03.01 – Информатика и вычислительная техника (ИиВТ) (уровень бакалавриата):

ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью.

Коды и наименования индикаторов достижения компетенции:

ОПК-4.1. Способен демонстрировать знание основных стандартов оформления технической документации на различных стадиях жизненного цикла информационной системы;

ОПК-4.2. Способен уметь применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы;

ОПК-4.3. Способен владеть навыками составления технической документации на различных этапах жизненного цикла информационной системы.

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

Коды и наименования индикаторов достижения компетенции:

ОПК-7.1. Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.

ОПК-7.2. Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.

ОПК-7.3. Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач.

В результате освоения дисциплины обучающийся должен:

Знать:

- методы описания синтаксических конструкций языков программирования, классификацию и эволюцию языков программирования;
- системы программирования (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:

- культурой мышления, умением аргументировано и ясно строить устную и письменную речь;
- основами профессиональной разговорной речи;
- 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 поиска и обмена информацией в глобальных и локальных компьютерных сетях.

Приобрести: опыт деятельности по разработке программ объемом до нескольких сотен операторов с использованием современных языков, методов и технологий программирования; опыт работы с современными операционными системами.

4. Содержание и структура дисциплины

В таблице 1 приводится описание содержания дисциплины, структурированное по разделам, с указанием по каждому разделу формы текущего контроля: защита лабораторной работы (ЛР), коллоквиум (К), рубежный контроль (РК), тестирование (Т).

Таблица 1

Содержание дисциплины

№ Раздела	Наименование раздела	Содержание раздела	Форма текущего контроля
1	Основные понятия языков программирования. Среды программирования	Основные конструкции языков программирования и их реализация. Методы разработки алгоритмов и программ. Технология программирования.	Коллоквиум (К), рубежный контроль (РК), тестирование (Т)
2	Базисные типы данных и операторный базис в процедурных языках программирования (на примере языка C++). Методы программирования с использованием статических структур	Данные и операторы. Классификация данных. Классификация операторов. Атрибуты данных и средства их описания. Характеристики, связанные с типом. Структура C++-программы. Базисные типы данных. Описания. Оператор присваивания. Организация ввода и вывода данных. Понятие исключительной ситуации. Операторы составной, условный, выбора. Числовые массивы. Блочная структура	Коллоквиум, рубежный контроль, тестирование

	данных	программы. Функции. 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, структуры, перечислимый тип. Типизированные константы. Строковый и символьный типы. Совместимость и преобразование типов. Комбинированный тип данных. Реализация методов упорядочения, поиска, выбора и слияния с использованием статических структур данных.	
3	Функции.	Структура функций. Область действия идентификаторов при использовании функций. Классификация способов передачи параметров. Передача параметров в C++.	Коллоквиум, рубежный контроль, тестирование
4	Рекурсивные методы в программировании.	Понятие рекурсии. Рекурсивный спуск. Рекурсивный возврат. Формы рекурсивных процедур. Перекрестная рекурсия.	Коллоквиум, рубежный контроль, тестирование
5	Понятие файла. Методы упорядочения файлов	Файлы. Доступ к файлам. Функции для работы с файлами. Текстовые файлы.	Коллоквиум, рубежный контроль, тестирование
6	Несвязанные динамические структуры данных	Динамическая память. Адреса и указатели. Объявление указателей. Выделение и освобождение динамической памяти. Использование указателей. Несвязанные динамические данные.	Коллоквиум, рубежный контроль, тестирование
7	Связанные динамические структуры данных. Методы программирования с использованием связанных динамических структур данных	Функции для работы с динамической памятью. Сложные структуры данных. Организация взаимосвязей в связанных динамических данных. Очереди. Набор базисных операций над очередями. Дисциплина обработки очередей. Стеки. Дисциплины обслуживания стеков. Занесение элемента в стек. Выбор элемента из стека. Списки. Виды списков. Дисциплина обработки списков. Двухнаправленные и кольцевые списки. Вставка, удаление и поиск элементов в	Коллоквиум, рубежный контроль, тестирование

		двунаправленных и кольцевых списках.	
8	Понятие и методы объектно-ориентированного программирования	<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. Наследование в ЯП. Понятия и примеры наследования.</p> <p>Наследование и полиморфизм. Объявление объектов. Конструкторы и деструкторы. Виртуальные методы. Абстрактные методы. Инкапсуляция.</p> <p>Другие парадигмы программирования: функциональное и логическое программирование.</p>	Коллоквиум, рубежный контроль, тестирование
9	Методы трансляции	<p>Основные этапы трансляции и основные понятия лексического анализа. 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>	Коллоквиум, рубежный контроль, тестирование
10	Основные концепции и методы параллельного и распределенного программирования.	<p>Параллельное программирование.</p> <p>Распределенное программирование</p>	Тестирование

Таблица 2

Структура дисциплины

Вид работы	Трудовое время, часы
	1 семестр
Общая трудовое время (в зачетных единицах)	3
Контактная работа (в часах):	108
Лекции (Л)	-
Практические занятия (ПЗ)	34
Семинарские занятия (СЗ)	-
Лабораторные работы (ЛР)	17
Самостоятельная работа (в часах), в том числе контактная работа:	48
Курсовой проект (КП), курсовая работа (КР)	-
Расчетно-графическое задание (РГЗ)	-
Реферат (Р)	-
Эссе (Э)	-
Самостоятельное изучение разделов	-

Контрольная работа (К)	–
Подготовка и прохождение промежуточной аттестации	9
Вид промежуточной аттестации	Зачет с оценкой

Общая трудоемкость дисциплины составляет 3 зачетные единицы (108 часов).

Таблица 3

Лабораторные работы

№ занятия	Наименование лабораторных работ
1.	Алгоритмы линейной структуры.
2.	Алгоритмы разветвляющейся структуры.
3.	Алгоритмы, реализуемые с помощью циклов типа ДЛЯ.
4.	Алгоритмы, реализуемые с помощью кратных циклов типа ДЛЯ.
5.	Алгоритмы, реализуемые с помощью циклов типа ПОКА.
6.	Алгоритмы, реализуемые с помощью кратных циклов типа ПОКА.
7.	Алгоритмы, реализуемые с помощью комбинаций циклов типа ДЛЯ и ПОКА.
8.	Функции.
9.	Символьный и строковый типы данных.
10.	Рекурсия.
11.	Файлы.
12.	Полиморфизм и перегрузка функций
13.	Шаблоны функций
14.	Динамическая структура типа стек
15.	Динамическая структура типа очередь
16.	Динамическая структура - двунаправленный список
17.	Работа с классами
18.	Перегрузка операций и дружественные функции
19.	Наследование классов

Практические занятия

Не предусмотрено.

Таблица 4

Самостоятельное изучение разделов дисциплины

№ п/п	Вопросы, выносимые на самостоятельное изучение
1.	Алгоритмы структурного программирования.
2.	Функции.
3.	Символьный и строковый типы данных.
4.	Рекурсия.
5.	Файлы.
6.	Работа с классами
7.	Полиморфизм и перегрузка функций Динамическая структура типа очередь

8.	Шаблоны функций
9.	Динамическая структура типа стек. инамическая структура - двунаправленный список
10.	Перегрузка операций и дружественные функции
11.	Наследование классов

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

Формы контроля текущих, рубежных и промежуточных знаний студентов по дисциплине определяются в соответствии с учебным планом образовательной программы и в соответствии с действующим Положением о балльно-рейтинговой системе оценки успеваемости студентов КБГУ.

От обучающихся требуется посещение занятий, выполнение лабораторных работ, знакомство с рекомендованной литературой.

При аттестации обучающихся оценивается качество работы на занятиях (умение вести дискуссию, способность четко и ёмко формулировать свои мысли), уровень подготовки к самостоятельной деятельности, качество выполнения заданий (презентаций, докладов, выполнение лабораторных работ и др.).

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

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 «Практикум по программированию» и включает: отчет по результатам выполнения лабораторных работ, самостоятельное выполнение индивидуальных домашних заданий с отчетом (защитой) в установленный срок.

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

Критерии формирования оценок (оценивания) устного опроса

Устный опрос является одним из основных способов учёта знаний обучающегося по дисциплине «Практикум по программированию». Развёрнутый ответ должен представлять собой связное, логически последовательное сообщение на заданную тему, показывать его умение применять определения.

В результате устного опроса знания, обучающегося оцениваются по следующей шкале:

Таблица 6

3 балла	2 балла	1 балл	0 баллов
ставится, если обучающийся:	ставится, если обучающийся даёт	ставится, если обучающийся	ставится, если обучающийся

1) полно излагает изученный материал, даёт правильное определенное экономических понятий; 2) обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные; 3) излагает материал последовательно и правильно с точки зрения норм литературного языка.	ответ, удовлетворяющий тем же требованиям, что и для балла «1», но допускает 1-2 ошибки, которые сам же исправляет, и 1-2 недочёта в последовательности и языковом оформлении излагаемого.	обнаруживает знание и понимание основных положений данной темы, но: 1) излагает материал неполно и допускает неточности в определении понятий; 2) не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; 3) излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.	обнаруживает незнание большей части соответствующего раздела изучаемого материала, допускает ошибки в формулировке.
--	--	--	---

Баллы «1», «2», «3» могут ставиться не только за единовременный ответ, но и за рассредоточенный во времени, т.е. за сумму ответов, данных на протяжении занятия. начисляются в зависимости от сложности задания.

5.1.1. Оценочные материалы для самостоятельной работы обучающегося

Рабочая программа предусматривает проведение лекционных, лабораторных занятий, а также самостоятельную работу обучающихся. В ФГБОУ ВО «Кабардино-Балкарский государственный университет» действует балльно-рейтинговая система оценки учебных достижений обучающихся по образовательным программам, реализуемым на основании федеральных государственных образовательных стандартов. Балльно-рейтинговая система оценки знаний является одной из составляющих системы управления качеством образовательной деятельности в университете.

Темы для самостоятельной работы

1. Алгоритмы структурного программирования.
2. Функции.
3. Символьный и строковый типы данных.
4. Рекурсия.
5. Файлы.
6. Работа с классами.
7. Полиморфизм и перегрузка функций.
8. Динамическая структура типа очередь.

9. Шаблоны функций.
10. Динамическая структура типа стек.
11. Динамическая структура - двунаправленный список.
12. Перегрузка операций и дружественные функции.
13. Наследование классов.

Критерии формирования оценок по заданиям для самостоятельной работы студента

«Отлично» (5 баллов) – обучающийся показал глубокие знания материала по поставленным вопросам, грамотно, логично его излагает, структурировал и детализировал информацию, избегая простого повторения информации из текста, информация представлена в переработанном виде. Свободно использует необходимые формулы при решении задач;

«Хорошо» (4 балла) – обучающийся твердо знает материал, грамотно его излагает, не допускает существенных неточностей в процессе решения задач;

«Удовлетворительно» (3 балла) – обучающийся имеет знания основного материала по поставленным вопросам, но не усвоил его деталей, допускает отдельные неточности при решении задач;

«Неудовлетворительно» (менее 3 баллов) – 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 используется тестирование (письменное или компьютерное), проведение коллоквиума.

Рубежный контроль освоения студентом дисциплины осуществляется в рамках балльно-рейтинговой системы. Распределение баллов в соответствии с действующим Положением о балльно-рейтинговой системе оценки успеваемости студентов КБГУ приведено в таблице 7.

Таблица 7

Распределение баллов в соответствии с действующим Положением о балльно-рейтинговой системе

№ рейтинговой точки	Коллоквиум	Лаб.практикум	Посещаемость	Тестирование	Итого
1	7	8	3	5	23
2	7	8	3	5	23
3	7	8	4	5	24

Таблица 8

Критерии оценки

Вид мероприятия	Критерии оценки	Баллы
Коллоквиум (устный опрос по теме)	- ясность, четкость и доказательность изложения ответов на вопросы; - владение специальными терминами; - системность знаний по тематике	0-21 балл
Лабораторное занятие	- понимание цели и задач работы	0-24 балла

	- выполнение заданий и обработка результатов - отчет и защита лабораторной работы	
Компьютерное тестирование по разделам дисциплины	Результаты тестирования (Количество баллов = $5 \cdot \varphi$, φ - доля правильно отвеченных тестов по теме).	0-15 баллов
Посещение занятий	При более 3 пропусках без уважительной причины занятий аннулируются баллы	0-10 баллов
Зачет	ясность, четкость и доказательность изложения ответов на вопросы; - владение специальными терминами; - системность знаний по тематике дисциплины в целом	0-30 баллов
Итоговая оценка		0-100 баллов

5.2.1. Оценочные материалы для проведения коллоквиума

Перечень вопросов, выносимых на рейтинговый контроль первой точки (контролируемые компетенции ОПК-7):

- 1) На какие группы можно разделить используемые в программировании типы данных?
- 2) Допустимо ли использование в качестве пользовательских идентификаторов зарезервированных слов C++?
- 3) Что представляет собой идентификатор?
- 4) Что представляет собой метка?
- 5) Какие формы записи чисел используются в языке C++?
- 6) Какой символ используется для обозначения шестнадцатеричных чисел?
- 7) Что представляет собой строка символов?
- 8) Что представляет собой комментарий?
- 9) Какие символы используются в качестве разделителя лексем?
- 10) Какое количество символов-разделителей допускается между любыми двумя лексемами?
- 11) Какими свойствами характеризуются порядковые типы?
- 12) Что представляют собой пользовательские типы?
- 13) Какие типы относятся к структурированным?
- 14) Что представляют собой типизированные константы?
- 15) Как происходит работа оператора Case?
- 16) Для чего предназначен перечислимый тип?
- 17) Опишите правила сравнения строк.
- 18) Какой тип характеризует объекты, называемые структурами?
- 19) Могут ли компоненты структуры быть различных типов?
- 20) Как осуществляется доступ к полям структуры?
- 21) Как упростить доступ к полям структуры?
- 22) Что является селектором в операторе Case при задании вариантной части записи?
- 23) Могут ли структуры входить компонентами в другие переменные?

Перечень вопросов, выносимых на рейтинговый контроль второй точки (контролируемые компетенции ОПК-7):

- 1) В каких случаях удобно использовать файлы?
- 2) Где хранятся файлы?
- 3) В чем состоит сходство и различие между массивом и файлом?
- 4) Как в разделе типов задается файловых тип?
- 5) Как подразделяются файлы по видам доступа к его компонентам?

- 6) Каким образом описываются переменные файловых типов?
- 7) Что необходимо выполнить для открытия файла?
- 8) Какие функции предназначены для открытия файлов и как они работают?
- 9) Как осуществляется доступ к компонентам файла?
- 10) Какие операции определены над файлами?
- 11) В чем состоят особенности текстовых файлов?
- 12) На какие группы можно разделить используемые в программировании типы данных?
- 13) В чем состоит основное отличие между данными статической и динамической структуры?
- 14) Какие данные относятся к данным статической структуры?
- 15) Какие данные относятся к однородным?
- 16) Какие данные относятся к неоднородным?
- 17) По каким признакам могут быть классифицированы массивы?
- 18) Какие данные относятся к данным динамической структуры?
- 19) Какие существуют разновидности связанных динамических данных?
- 20) В чем особенность объявления данных динамической структуры?
- 21) С помощью каких процедур происходит распределение памяти под динамические переменные?
- 22) Что означает состояние null указателя?
- 23) Какие действия выполняют функции New и Delete?

**Перечень вопросов, выносимых на рейтинговый контроль третьей точки
(контролируемые компетенции ОПК-7):**

- 1) В чем выражаются динамические свойства несвязанных динамических данных?
- 2) В чем сходство и различие между линейными и кольцевыми списками?
- 3) Что представляет собой очередь как структура данных?
- 4) Что представляет собой стек как структура данных?
- 5) Что представляет собой пирамида как структура данных и в чем ее отличие от дерева?
- 6) Что требуется для создания связанных динамических структур данных?
- 7) Сколько указателей требуется для работы с очередью?
- 8) Какие действия необходимо выполнить для работы с очередью?
- 9) Как добавить или удалить элемент очереди?
- 10) Какие действия необходимо выполнить для работы со стеком?
- 11) Как добавить или удалить элемент стека?
- 12) Сколько указателей требуется для работы со стеком?
- 13) В чем суть объектно-ориентированной методологии программирования (ООП)?
- 14) Каково назначение ООП?
- 15) Дайте определение понятию «объект» в ООП.
- 16) Опишите разновидности декомпозиции: алгоритмическую и объектно-ориентированную. Приведите примеры каждого из видов декомпозиции.
- 17) В чем суть принципов инкапсуляции, наследования и полиморфизма?
- 18) Что такое «абстрактные типы»? На каких уровнях ОО-иерархии они обычно размещаются?
- 19) Опишите назначение ключевых слов private и protected.
- 20) Какие методы называются виртуальными? Каково основное назначение виртуальных методов?
- 21) Как объявляются виртуальные методы?
- 22) Что такое конструкторы и деструкторы?
- 23) Как формируется таблица виртуальных методов?
- 24) Опишите две основные возможности модификации программ при использовании виртуальных методов.
- 25) На чем основан рекурсивный метод программирования?
- 26) В чем состоит мощь рекурсивных определений?
- 27) В чем преимущества и недостатки использования рекурсивных процедур по сравнению с нерекурсивными?

- 28) Что называется «глубиной рекурсии»?
- 29) Какой элемент является обязательным в рекурсивном определении?
- 30) К каким последствиям приводит «рекурсивное заикливание»?
- 31) Какое условие должно обязательно присутствовать в любой рекурсивной процедуре?
- 32) Что такое явная и косвенная рекурсии?
- 33) Дайте рекурсивное определение целой степени числа N.
- 34) Почему простой рекурсивный алгоритм для вычисления чисел Фибоначчи неэффективен?
- 35) Можно ли утверждать, что нерекурсивный алгоритм для вычисления факториала намного эффективнее рекурсивного?

Критерии формирования оценок по контрольным точкам (коллоквиум)

4 балла ставится за работу, выполненную полностью без ошибок и недочетов; обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, решено 100% задач;

3 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;

2 балла ставится за работу, если обучающийся 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

1 балл ставится за работу, если число ошибок и недочетов превысило норму или правильно выполнено менее 2/3 всей работы. Обучающийся дает неверную оценку ситуации, решено менее 50 % задач.

5.2.2. Оценочные материалы: тестирование

Тестирование обучающихся проводится в онлайн-режиме согласно расписанию в ЭИОС open.kbsu.ru

Примерные тестовые задания (контролируемые компетенции ОПК-7)

Полный перечень тестовых заданий представлен в ЭОИС

1. Результатом выполнения программы

```
#include <stdio.h>
#define _N    20
void main()
{
    #if _N + 20
    int n=5; printf("\n %d", (--n)*_N);
    #else
    int n=10; printf("\n %d", (n++)*_N);
    #endif
}
```

будет

-: 100

+: 80

-: 220

-: ничего

2. Результатом выполнения программы

```
#include <stdio.h>
```

```
#define N 3
```

```
#define X(k) x[k-1]
```

```
void main(){
```

```
    int *x; x=new[N] ;
```

```
    for(int i=1;i<=N;i++) X(i)=i;
```

```
    printf("\n %d%d%d", X(1),X(2),X(3) );
```

```
    delete (x);
```

```
}
```

будет

-: 100

-: 012

-: 120

+: 123

3. Результатом выполнения программы

```
#include <stdio.h>
```

```
#define N 3
```

```
#define X(p, q) x[p*N + q]
```

```
void main(){
```

```
    int *x, i, j; // выделяется память//
```

```
    for(i =0; i< N; i++) for(j=0;j<N;j++) X(i,j)=i +j;
```

```
    printf("\n %d%d%d", X(1,0),X(1,1),X(1,2) );
```

```
    // память освобождается//
```

```
}
```

будет

-: 012

+: 123

-: 021

-: 120

4. Директивы препроцессора (#include, #define и др.)

+: управляют преобразованием текста до компиляции программы.

- : управляют преобразованием текста после компиляции программы.
- : управляют преобразованием текста во время компиляции.
- : ничего не делают.

5. Правильную последовательность типов данных по возрастанию области измерения составляют

- +: float , double , long double
- : double , long double , float
- : long double , double , float
- : float , long double , double

6. Приставка 0x в фрагменте `int x= 0x27;` означает

- : то, что 27 записано в восьмеричной системе счисления
- : то, что переменная x это указатель на число 27
- +: то, что 27 записано в шестнадцатеричной системе счисления
- : то, что переменная x перегруженный указатель на число 27

7. Множественное наследование означает, что

- +: Класс имеет несколько базовых классов
- : Класс имеет несколько виртуальных методов
- : Класс имеет дружественные функции
- : Иное

8. Когда один класс наследуется из другого класса, то наследуемый класс называется

- : базовым
- +: производным
- : интерпретируемым
- : виртуальным

9. Деструктор

- +: Вызывается для удаления объектов из памяти
- +: Вызывается автоматически при использовании оператора delete
- : Используется для освобождения области памяти при создании объекта
- : Функция, которая вызывается при создании объекта

10. Класс - это:

- +: Пользовательский тип данных
- +: Данные и методы их обработки
- : Объявленный определенным способом массив структур
- : Объект имеющий свою область памяти для размещения используемых им данных

Критерии формирования оценок по тестовым заданиям

Выполнение тестирования оценивается согласно проценту правильных ответов. Максимально возможное количество баллов за тестирование – 5.

5.3. Оценочные материалы для промежуточной аттестации

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 «Практикум по программированию» в виде проведения зачета в 3 семестре и экзамена в 4 семестре.

Промежуточная аттестация может проводиться в устной, письменной форме, и в форме тестирования. На промежуточную аттестацию отводится до 30 баллов.

Перечень вопросов для подготовки к зачету (контролируемые компетенции ОПК-7)

1. Язык C++. Алфавит. Символы, используемые в идентификаторах. Разделители. Специальные символы (знаки пунктуации, знаки операций, зарезервированные слова).
2. Язык C++. Простые типы данных: целые типы, логический тип, символьный тип, перечислимый тип, тип-диапазон, тип-указатель, вещественные типы.
3. Язык C++. Множественный тип значений. Задание множественного типа и множественные переменные. Операции над множествами. Множественные выражения. Примеры.
4. Язык C++. Блочная структура программы. Порядок размещения разделов объявлений. Принцип локализации переменных. Глобальные и локальные идентификаторы. Примеры.
5. Язык C++. Стандартные функции: арифметические (abs, exp, sign, sin, ln, ...); преобразования типов. Строковые функции (delete, insert, str, val, concat, copy, length, pos).
6. Язык C++. Массивы. Определение массива. Примеры описаний массивов, составленных из чисел, символов, строк.
7. Язык C++. Оператор присваивания арифметический, логический, литерный. Примеры.
8. Язык C++. Условный оператор IF. Общий вид. Полная и неполная формы. Вложенные условные операторы. Использование в условном операторе составного оператора. Примеры.
9. Язык C++. Оператор варианта. Назначение. Общий вид. Примеры.
10. Язык C++. Оператор цикла FOR. Назначение. Общий вид. Примеры.
11. Язык C++. Оператор цикла WHILE. Назначение. Общий вид. Примеры.
12. Язык C++. Процедуры без параметров. Глобальные и локальные переменные.
13. Язык C++. Процедуры-операторы. Назначение, синтаксис, описание, активация (вызов). Формальные и фактические параметры. Параметры-значения. Параметры-переменные. Параметры-массивы. Глобальные и локальные переменные.
14. Язык C++. Процедуры-функции. Назначение, синтаксис, описание, активация (вызов). Формальные и фактические параметры. Глобальные и локальные переменные. Параметры-значения. Параметры-переменные. Параметры-массивы. Побочные эффекты функции. Примеры.
15. Среда программирования Visua Studio. Вход в интегрированную среду. Окна редактирования. Окна диалога. Информационные окна. Система меню среды. Меню работы с файлами (File). Меню редактирования (Edit). Меню поиска информации (Search). Меню выполнения программы (Run). Меню компиляции (Compile). Меню параметров среды (Options). Меню окон (Window). Меню информационной помощи (Help)

16. Суть рекурсивного метода программирования. Преимущества и недостатки использования рекурсивных подпрограмм по сравнению с нерекурсивными. Глубина рекурсии. Якорная ветвь рекурсивной подпрограммы. Проблема рекурсивного заикливания. Явная и косвенная рекурсия.
17. Статические и динамические программные объекты. Принципы их порождения, время их существования.
18. Распределение памяти для глобальных переменных и локальных переменных процедур. Размеры сегментов памяти, выделяемых программе и статическим объектам. Размер динамической области памяти.
19. Задание ссылочного типа переменных. Действия над ссылками. Пустая ссылка. Типизированные и нетипизированные указатели.
20. Выделение и освобождение динамической памяти. Куча. Назначение. Примеры. Процедуры и функции для работы с динамической памятью. Примеры использования.
21. Статические и динамические программные объекты. Принципы их порождения, возможность изменения в процессе работы программы объемов памяти, необходимых для их размещения, время их существования.
22. Распределение памяти для глобальных переменных и локальных переменных процедур. Размеры сегментов памяти, выделяемых программе и статическим объектам. Размер динамической области памяти.
23. Задание ссылочного типа переменных. Действия над ссылками. Пустая ссылка. Типизированные и нетипизированные указатели.
24. Стеки. Стеки FIFO и LIFO. Дисциплины обслуживания стеков. Занесение элемента в стек. Выбор элемента из стека. Примеры использования стеков.
25. Абстрактный тип списков. Виды списков. Дисциплина обработки списков. Примеры.
26. Динамическая структура двунаправленного списка. Кольцевой список. Вставка, удаление и поиск элементов в двунаправленном и кольцевом списках. Примеры.
27. Очереди. Набор базисных операций над очередями. Дисциплина обработки очередей. Примеры.
28. Объектно-ориентированное программирование. Инкапсуляция. Наследование. Полиморфизм. Виртуальные методы. Объявление виртуальных методов. Конструкторы и деструкторы. Пример использования ООП.
29. Классическая структура компилятора, ее составляющие.
30. Этапы процесса компиляции.
31. Основные концепции параллельного и распределенного программирования. Параллельная обработка данных. Конвейерная обработка данных. Многопроцессорная обработка. Суперкомпьютеры: векторно-конвейерные, массивно-параллельные с распределенной памятью, параллельные с общей памятью. Пиковая производительность компьютера. Закон Амдала и его следствия. Анализ узких мест в архитектуре суперкомпьютеров.

Методические рекомендации при подготовке к зачету с оценкой

Подготовка студентов к зачету включает проработку лекций, в течение семестра и непосредственную подготовку в дни, предшествующие экзамену, включая, конечно, подготовку к коллоквиумам, тестированию, выполнению лабораторных работ и их защите.

Для подготовки к ответам на зачетные вопросы (они выдаются в конце семестра) студент должен использовать не только курс лекций, но и основную и дополнительную литературу для выработки умения давать развернутые ответы на поставленные вопросы.

В ходе подготовки к э зачету студенту необходимо обращать внимание не только на уровень запоминания, но и на степень понимания изучаемых вопросов. А это достигается не простым заучиванием, а усвоением прочных систематизированных знаний аналитическим мышлением. Следовательно, непосредственная подготовка к зачету должна в разумных пропорциях сочетать и запоминание, и понимание программного материала.

Критерии оценивания

Оценка			
неудовлетворительно 0 баллов	удовлетворительно 3 балла	хорошо 4 балла	отлично 5 баллов
Посещение менее 50 % лекционных и практических занятий.	Посещение не менее 60% лекционных и практических занятий.	Посещение не менее 70 % лекционных и практических занятий.	Посещение не менее 85% лекционных и практических занятий.
Студент не знает значительной части материала, допускает существенные ошибки в ответах на вопросы.	Студент поверхностно знает материал основных разделов и тем учебной дисциплины, допускает неточности в ответе на вопрос.	Студент хорошо знает материал, грамотно и по существу излагает его, допуская некоторые неточности в ответе на вопрос.	Студент в полном объеме знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос.

6. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

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 «Практикум по программированию» является зачет с оценкой.

Общий балл текущего и рубежного контроля складывается из следующих составляющих (Приложение).

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

Таблица 9

Результаты освоения учебной дисциплины, подлежащие проверке

Результаты обучения (компетенции)	Основные показатели оценки результатов обучения	Вид оценочного материала, обеспечивающего формирование компетенций
ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1. Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. ОПК-7.2. Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.	Типовые оценочные материалы для устного опроса, типовые тестовые задания, лабораторные работы, задания для самостоятельной работы (раздел 5)

	ОПК-7.3. Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач.	
--	--	--

7. Учебно-методическое обеспечение дисциплины

7.1. Нормативно-правовая база

1. ГОСТ «Единая система программной документации».
2. ГОСТ Р ИСО/МЭК ТО 15271-2002 «Руководство по применению ГОСТ Р ИСО/МЭК 12270 (Процессы жизненного цикла программных средств)».
3. ГОСТ Р ИСО/МЭК ТО 12182-2010 «Классификация программных средств».
4. ISO/IEC 14764:2006 «Разработка программного обеспечения. Процессы жизненного цикла программного обеспечения. Сопровождение».
5. ГОСТ Р ИСО/МЭК 12119-2000 «Информационная технология. Пакеты программ. Требования к качеству и тестирование»
6. ГОСТ 28195-89 «Оценка качества программных средств. Общие положения».
7. ISO/IEC 25000:2005 «Технология программного обеспечения. Требования и оценка качества программного продукта. Руководство».
8. ISO/IEC 25001:2014 «Программирование. Требования к качеству программного продукта и его оценка. Планирование и менеджмент».
9. ISO/IEC 25010:2011 «Проектирование систем и разработка программного обеспечения. Требования к качеству систем и программного обеспечения и их оценка (SQuaRE). Модели качества систем и программного обеспечения».
10. ISO/IEC 25012:2008 «Программная инженерия – Требования к качеству и оценке программного обеспечения. Модель качества данных».
11. ISO/IEC 25020:2007 «Разработка программного обеспечения. Требования к качеству и оценка качества программного продукта. Измерительная эталонная модель и руководство».

7.2. Основная литература

1. Костюкова Н.И. Программирование на языке Си [Электронный ресурс]: методические рекомендации и задачи по программированию/ Костюкова Н.И.— Электрон. текстовые данные.— Новосибирск: Сибирское университетское издательство, 2017.— 160 с.— Режим доступа: <http://www.iprbookshop.ru/65289.html>.— ЭБС «IPRbooks»
2. Программирование на языке высокого уровня C/C++ [Электронный ресурс]: конспект лекций/ — Электрон. текстовые данные.— М.: Московский государственный строительный университет, Ай Пи Эр Медиа, ЭБС АСВ, 2016.— 140 с.— Режим доступа: <http://www.iprbookshop.ru/48037.html>.— ЭБС «IPRbooks»
3. Стенли Липпман Язык программирования C++ [Электронный ресурс]: полное руководство/ Стенли Липпман, Жози Лажойе— Электрон. текстовые данные.— Саратов: Профобразование, 2017.— 1104 с.— Режим доступа: <http://www.iprbookshop.ru/63964.html>.— ЭБС «IPRbooks»
4. Тюгашев А.А. Основы программирования. Часть 1 [Электронный ресурс]: учебное пособие/ Тюгашев А.А.— Электрон. текстовые данные.— СПб.: Университет ИТМО, 2016.— 163 с.— Режим доступа: <http://www.iprbookshop.ru/67495.html>.— ЭБС «IPRbooks»
5. Тюгашев А.А. Основы программирования. Часть 2 [Электронный ресурс]: учебное пособие/ Тюгашев А.А.— Электрон. текстовые данные.— СПб.: Университет

ИТМО, 2016.— 120 с.— Режим доступа: <http://www.iprbookshop.ru/67496.html>.— ЭБС «IPRbooks»

7.3. Дополнительная литература

1. Котов О.М. Язык C#. Краткое описание и введение в технологии программирования [Электронный ресурс]: учебное пособие/ Котов О.М.— Электрон. текстовые данные.— Екатеринбург: Уральский федеральный университет, 2014.— 208 с.— Режим доступа: <http://www.iprbookshop.ru/68524.html>.— ЭБС «IPRbooks»
2. Липпман Стенли Б., Лажоие Жози, Му Барбара Э. Язык программирования C++. Базовый курс. М.: Вильямс, 2014. – 1120 с. – 5-е изд.
3. Поляков А.Ю. Программирование [Электронный ресурс]: практикум/ Поляков А.Ю., Полякова А.Ю., Перышкова Е.Н.— Электрон. текстовые данные.— Новосибирск: Сибирский государственный университет телекоммуникаций и информатики, 2015.— 55 с.— Режим доступа: <http://www.iprbookshop.ru/55494.html>.— ЭБС «IPRbooks»

7.4. Периодические издания (газета, вестник, бюллетень, журнал)

1. Журнал «Объектно-ориентированное программирование для профессионалов».
2. Журнал «Компьютеры & Программы».
3. Журнал «Программирование».

7.5. Интернет-ресурсы

1. <http://iprbookshop.ru>
2. <https://www.codecademy.com/>
3. <https://itvdn.com/ru>
4. <http://lendwings.com/>

7.6. Современные профессиональные базы данных

1. База данных Science Index (РИНЦ) <http://elibrary.ru>
2. Национальная электронная библиотека РГБ <https://нэб.рф>
3. Крупнейшая единая база данных, содержащая аннотации и информацию о цитируемости рецензируемой научной литературы, со встроенными инструментами отслеживания, анализа и визуализации данных. www.scopus.com
4. Самая полная математическая база данных, охватывающая материалы с конца 19 века. zbMath содержит документы, журналы и книги по математике, статистике, информатике, а также машиностроению, физике, естественным наукам и др. www.zbmath.org (доступ открытый).

7.7. Методические указания по проведению различных учебных занятий и самостоятельной работы

Методические указания к лабораторным занятиям

Самостоятельная подготовка студентов к работе. Перед началом работы студенты должны четко представлять себе цель работы, знать схему, метод измерения, физическую сущность ожидаемых результатов. Должен быть подготовлен отчет, содержащий о порядке выполнения лабораторной работы. Студенты, не подготовившиеся к работе в соответствии с этими требованиями, к выполнению работы не допускаются.

Проведение эксперимента. Этот этап осуществляется в соответствии с методическими указаниями, которые содержатся в описании к каждой работе. Приступать к работам на стенде студент может начать только после ознакомления с теоретической частью и описания хода выполнения работы. Любые изменения в схеме проводятся при тщательной проверке схемы, для исключения короткого замыкания. Результаты выполнения проверяются преподавателем.

Составление отчета о проделанной работе. Отчёт должен содержать исчерпывающие данные, как о цели работы, так и о результатах в следующей последовательности: задание; схема установки и описание хода выполнения; результаты выполнения работы, включая рисунки, схемы, таблицы; общие выводы и заключение.

Текст отчета должен быть написан аккуратно и разборчиво от руки или представлен в виде распечатки, после компьютерной верстки. В обоих случаях текст должен представлять собой логическое изложение существа вопроса. Недопустимо приведение формул, таблиц без разъяснений всех обозначений и сокращений. Полученные зависимости должны сопровождаться теоретическим обоснованным объяснением причин, влияющих на их ход, для чего в процессе составления отчета студент обязан по литературным источникам ознакомиться с материалом, который был объектом его исследования в лаборатории. Без такого ознакомления с испытуемым методом студент не будет в состоянии дать правильный анализ процессов, происходящих в материале при эксперименте.

Защита лабораторной работы с представлением отчета. При сдаче отчета студенты должны показать понимание сущности проведенных исследований, объяснить полученные результаты и сделать выводы. При работе в лаборатории необходимо строго выполнять все правила техники безопасности и указания преподавателя.

Методические рекомендации по организации самостоятельной работы

Самостоятельная работа обучающихся – способ активного, целенаправленного приобретения обучающимися 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 обучающихся в подборе материала, поиске путей решения задач.

Самостоятельная работа приводит обучающихся 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:

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

Обучающимся 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 обучающийся получает в аудитории.

Необходимо отметить, что некоторые задания для самостоятельной работы по курсу имеют определенную специфику. При освоении курса обучающийся 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 обучающихся предусмотрена учебным планом и выполняется в обязательном порядке. Задания предложены по каждой изучаемой теме и могут готовиться индивидуально или в группе. По необходимости обучающийся может обращаться за консультацией к преподавателю. Выполнение заданий контролируется и оценивается преподавателем.

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 обучающихся и позволяет кардинальным образом изменить методику преподавания.

Обучающийся может получать все задания и методические указания через сервер, что дает ему возможность привести в соответствие личные возможности с необходимыми для выполнения работ трудозатратами. Обучающийся имеет возможность выполнять работу дома или в аудитории. Большое воспитательное и образовательное значение в самостоятельном учебном труде обучающийся имеет самоконтроль. Самоконтроль возбуждает и поддерживает внимание и интерес, повышает активность памяти и мышления, позволяет обучающемуся 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 прочесть 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.

8. Материально-техническое обеспечение дисциплины

8.1. Требования к материально-техническому обеспечению

Минимально необходимый для реализации ОПОП перечень материально-технического обеспечения включает в себя: лекционные аудитории (оборудованные видеопроекционным оборудованием для презентаций, средствами звуковоспроизведения, экраном и имеющие выход в сеть Интернет), помещения для проведения семинарских и практических занятий (оборудованные учебной мебелью), компьютерные классы и др. (в соответствии с ФГОС и учебным планом).

Во время самостоятельной работы студенты используют компьютерные классы института информатики, электроники и компьютерных технологий, электронные читальные залы КБГУ и домашние компьютеры.

Лицензионное программное обеспечение, используемое для проведения лекционных и лабораторных занятий

1. Microsoft Windows 7, 8, 10 Enterprise.
2. Microsoft Office 2010, 2013, 2016 Professional.
3. Kaspersky Endpoint Securite 10 Standart.
4. Visual Studio.

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

Для студентов 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 (аудитория для самостоятельной работы и коллективного пользования специальными техническими средствами для обучения инвалидов и лиц с ОВЗ в КБГУ, аудитория № 145 Главный корпус КБГУ):

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 студентов 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 студенту 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 студента 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 рабочую программу по дисциплине «Практикум по программированию» по направлению подготовки 09.03.03 Прикладная информатика, Профиль Корпоративные информационные системы на _____ учебный год

[illegible]

Обсуждена и рекомендована на заседании кафедры информатики и информационной безопасности протокол № _____ от « _____ » _____ 201_____ г.

Заведующий кафедрой / /

Приложение

Распределение баллов текущего и рубежного контроля

№п/ п	Вид контроля	Сумма баллов			
		Общая сумма	1-я точка	2-я точка	3-я точка
1	Посещение занятий	до 10 б.	до 3 б.	до 3б.	до 4б.
2	Текущий контроль:	до 28 б.	до 9 б.	до 9 б.	до 10 б.
	Выполнение лабораторных работ	до 18 б.	до 6 б.	до 6 б.	до 6 б.
	Выполнение самостоятельных заданий	от 0 до 10 б.	от 0 до 3 б.	от 0 до 3 б.	от 0 до 4 б.
3	Рубежный контроль	до 27 баллов	до 9 б.	до 9 б.	до 9 б.
	тестирование	от 0 до 15 б.	от 0 до 5 б.	от 0 до 5 б.	от 0 до 5 б.
	коллоквиум	от 0 до 12 б.	от 0 до 4 б.	от 0 до 4 б.	от 0 до 4 б.
4	Итого сумма текущего и рубежного контроля	до 70 б.	до 23 б.	до 23 б.	до 24 б.