

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
КАБАРДИНО-БАЛКАРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМ. Х.М. БЕРБЕКОВА**

**ИНСТИТУТ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА И ЦИФРОВЫХ
ТЕХНОЛОГИЙ**

**КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ИНФОРМАЦИОННОЙ
БЕЗОПАСНОСТИ**

СОГЛАСОВАНО
Руководитель ОПОП

_____ Ф.Р. Кетова

« ____ » _____ 2024 г.

УТВЕРЖДАЮ
Директор ИИИиЦТ

_____ З.В. Шомахов

« ____ » _____ 2024 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Программирование на языке высокого уровня»

по направлению подготовки
09.03.03 – Прикладная информатика

Профиль:
Корпоративные информационные системы

Квалификация (степень) выпускника
Бакалавр

Очная форма
(форма обучения)

Нальчик 2024

Рабочая программа дисциплины «Программирование на языке высокого уровня»
/сост. Р.В. Гурфова – Нальчик: ФГБОУ КБГУ, 2024. – 23 с.

Рабочая программа предназначена для преподавания дисциплины вариативной части дисциплин по выбору студентам очной формы обучения по направлению подготовки 09.03.03 Прикладная информатика в 4 семестре.

Рабочая программа составлена с учетом федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.03. «Прикладная информатика» утвержденного приказом Министерства образования и науки Российской Федерации «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 содержания дисциплины.....	4
4. Содержание и структура дисциплины	5
5. Оценочные материалы для текущего и рубежного контроля успеваемости и промежуточной аттестации.....	9
6. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций	15
7. Учебно-методическое обеспечение дисциплины	17
8. Материально-техническое обеспечение дисциплины.....	22
9. Лист переутверждения рабочей программы дисциплины (модуля) Ошибка!	
Закладка не определена.	

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 задаче 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 относится к вариативной части дисциплин по выбору ОПОП ВО направления подготовки 09.03.03 «Прикладная информатика» профиля «Корпоративные информационные системы».

3. Требования к результатам освоения содержания дисциплины

В результате освоения дисциплины должны быть сформированы следующие компетенции:

ПКС-1 Способность проводить обследование организаций, выявлять информационные потребности пользователей, формировать требования к информационной системе

ПКС-2 Способность разрабатывать и адаптировать прикладное программное обеспечение

ПКС-6 Способность принимать участие в организации ИТ-инфраструктуры и управлении информационной безопасностью

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:

- 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 разделов дисциплины

№ Раздела	Наименование раздела	Содержание раздела	Контролируемые компетенции	Форма текущего контроля
1	2	3	4	5
1 семестр				
1	Введение.		ПКС-1; ПКС-2;	Коллоквиум (К),

	Основные понятия языков программирования. Среда программирования	Исторический очерк развития языков программирования (ЯП). Основные конструкции языков программирования и их реализация. Методы разработки алгоритмов и программ. Технология программирования.	ПКС-6	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 данных	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 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.	ПКС-1; ПКС-2; ПКС-6	Коллоквиум, рубежный контроль, тестирование
3	Функции.	Структура функций. Область действия идентификаторов при использовании функций. Классификация способов передачи параметров. Передача параметров в C++.	ПКС-1; ПКС-2; ПКС-6	Коллоквиум, рубежный контроль, тестирование
4	Рекурсивные методы в программировании.	Понятие рекурсии. Рекурсивный спуск. Рекурсивный возврат. Формы рекурсивных процедур. Перекрестная рекурсия.	ПКС-1; ПКС-2; ПКС-6	Коллоквиум, рубежный контроль, тестирование
5	Понятие файла. Методы упорядочения файлов	Файлы. Доступ к файлам. Функции для работы с файлами. Текстовые файлы.	ПКС-1; ПКС-2; ПКС-6	Коллоквиум, рубежный контроль, тестирование
6	Несвязанные динамические	Динамическая память. Адреса и указатели. Объявление указателей.	ПКС-1; ПКС-2; ПКС-6	Коллоквиум, рубежный

	структуры данных	Выделение и освобождение динамической памяти. Использование указателей. Несвязанные динамические данные.		контроль, тестирование
7	Связанные динамические структуры данных. Методы программирования с использованием связанных динамических структур данных	<p>Функции для работы с динамической памятью. Сложные структуры данных. Организация взаимосвязей в связанных динамических данных.</p> <p>Очереди. Набор базисных операций над очередями. Дисциплина обработки очередей.</p> <p>Стеки. Дисциплины обслуживания стеков. Занесение элемента в стек. Выбор элемента из стека.</p> <p>Списки. Виды списков. Дисциплина обработки списков.</p> <p>Двунаправленные и кольцевые списки. Вставка, удаление и поиск элементов в двунаправленных и кольцевых списках.</p>	ПКС-1; ПКС-2; ПКС-6	Коллоквиум, рубежный контроль, тестирование
8	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. Классы.</p> <p>Наследование в ЯП. Понятия и примеры наследования.</p> <p>Наследование и полиморфизм.</p> <p>Объявление объектов. Конструкторы и деструкторы. Виртуальные методы.</p> <p>Абстрактные методы. Инкапсуляция.</p> <p>Другие парадигмы программирования: функциональное и логическое программирование.</p>	ПКС-1; ПКС-2; ПКС-6	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.</p> <p>Лексический анализ для выражений в ОПЗ. Синтаксический анализ.</p> <p>Грамматика для арифметических выражений.</p>	ПКС-1; ПКС-2; ПКС-6	Коллоквиум, рубежный контроль, тестирование
10	Основные концепции и методы параллельного и распределенного программирования.	<p>Параллельное программирование.</p> <p>Распределенное программирование</p>	ПКС-1; ПКС-2; ПКС-6	Тестирование

Таблица 2. Общая трудоемкость дисциплины составляет 3 з.е. (108 часов)

Вид работы	Трудоемкость, часов	
	семестр	Всего
Общая трудоемкость:	108	108
Контактная работа:	51	51
<i>Лекции (Л)</i>	17	17
<i>Практические занятия (ПЗ)</i>	-	-
<i>Лабораторные работы (ЛР)</i>	34	34
Самостоятельная работа:	57	57
<i>Расчетно-графическое задание (РГЗ)</i>	-	-
<i>Реферат (Р)</i>	-	-
<i>Эссе (Э)</i>	-	-
<i>Контрольная работа (К)</i>	-	-
<i>Самостоятельное изучение разделов</i>	48	48
Курсовая работа (КР)/ Курсовой проект (КП)	-	-
Подготовка и прохождение промежуточной аттестации	9	9
Вид промежуточной аттестации	зачет	зачет

Таблица 3

Лекционные занятия

№ раз-дела	Наименование разделов
1	Введение. Основные понятия языков программирования. Среды программирования
2	Базисные типы данных и операторный базис в процедурных языках программирования (на примере языка C++). Методы программирования с использованием статических структур данных
3	Функции.
4	Рекурсивные методы в программировании.
5	Понятие файла. Методы упорядочения файлов

Таблица 4

Лабораторные работы

№ ЛР	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. Рекурсивные функции.

Практические занятия

Не предусмотрены.

Таблица 5

Самостоятельное изучение разделов дисциплины

№ раздела	Вопросы, выносимые на самостоятельное изучение
1	2
1	Форматированный вывод.
2	Рекурсия в C++.
3	Шаблоны функций.
4	Потоковый ввод/вывод.
5	Указатели. Динамические массивы.
6	Перегрузка операций.
7	Шаблоны классов.

Курсовой проект (курсовая работа)

Не предусмотрено.

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

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 «Программирование на ЯВУ» 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.

5.1.1. Оценочные материалы для самостоятельной работы обучающегося

Перечень типовых заданий для самостоятельной работы сформирован в соответствии с тематикой лабораторных занятий по дисциплине «Программирование на ЯВУ».

Темы для самостоятельной работы

1. Форматированный вывод.
2. Рекурсия в C++.

3. Шаблоны функций.
4. Поточный ввод/вывод.
5. Указатели. Динамические массивы.
6. Перегрузка операций.
7. Шаблоны 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.

5.2.1. Оценочные материалы для проведения коллоквиума

Вопросы рубежного контроля

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?
24. В каких случаях удобно использовать файлы?
25. Где хранятся файлы?
26. В чем состоит сходство и различие между массивом и файлом?
27. Как в разделе типов задается файловых тип?
28. Как подразделяются файлы по видам доступа к его компонентам?
29. Каким образом описываются переменные файловых типов?
30. Что необходимо выполнить для открытия файла?
31. Какие функции предназначены для открытия файлов и как они работают?
32. Как осуществляется доступ к компонентам файла?
33. Какие операции определены над файлами?
34. В чем состоят особенности текстовых файлов?
35. На какие группы можно разделить используемые в программировании типы данных?
36. В чем состоит основное отличие между данными статической и динамической структуры?
37. Какие данные относятся к данным статической структуры?
38. Какие данные относятся к однородным?
39. Какие данные относятся к неоднородным?
40. По каким признакам могут быть классифицированы массивы?
41. Какие данные относятся к данным динамической структуры?
42. Какие существуют разновидности связанных динамических данных?
43. В чем особенность объявления данных динамической структуры?
44. С помощью каких процедур происходит распределение памяти под динамические переменные?
45. Что означает состояние null указателя?
46. Какие действия выполняют функции New и Delete?
47. В чем выражаются динамические свойства несвязанных динамических данных?
48. В чем сходство и различие между линейными и кольцевыми списками?
49. Что представляет собой очередь как структура данных?
50. Что представляет собой стек как структура данных?
51. Что представляет собой пирамида как структура данных и в чем ее отличие от дерева?
52. Что требуется для создания связанных динамических структур данных?
53. Сколько указателей требуется для работы с очередью?
54. Какие действия необходимо выполнить для работы с очередью?
55. Как добавить или удалить элемент очереди?
56. Какие действия необходимо выполнить для работы со стеком?
57. Как добавить или удалить элемент стека?
58. Сколько указателей требуется для работы со стеком?
59. В чем суть объектно-ориентированной методологии программирования (ООП)?
60. Каково назначение ООП?
61. Дайте определение понятию «объект» в ООП.
62. Опишите разновидности декомпозиции: алгоритмическую и объектно-ориентированную. Приведите примеры каждого из видов декомпозиции.

63. В чем суть принципов инкапсуляции, наследования и полиморфизма?
64. Что такое «абстрактные типы»? На каких уровнях ОО-иерархии они обычно размещаются?
65. Опишите назначение ключевых слов `private` и `protected`.
66. Какие методы называются виртуальными? Каково основное назначение виртуальных методов?
67. Как объявляются виртуальные методы?
68. Что такое конструкторы и деструкторы?
69. Как формируется таблица виртуальных методов?
70. Опишите две основные возможности модификации программ при использовании виртуальных методов.
71. На чем основан рекурсивный метод программирования?
72. В чем состоит мощьность рекурсивных определений?
73. В чем преимущества и недостатки использования рекурсивных процедур по сравнению с нерекурсивными?
74. Что называется «глубиной рекурсии»?
75. Какой элемент является обязательным в рекурсивном определении?
76. К каким последствиям приводит «рекурсивное заикливание»?
77. Какое условие должно обязательно присутствовать в любой рекурсивной процедуре?
78. Что такое явная и косвенная рекурсии?
79. Дайте рекурсивное определение целой степени числа N .
80. Почему простой рекурсивный алгоритм для вычисления чисел Фибоначчи неэффективен?
81. 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 системе ЭИОС open.kbsu.

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 «Программирование на ЯВУ» в виде проведения экзамена.

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

Вопросы к зачету

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++. Оператор цикла REPEAT. Назначение. Общий вид. Примеры.
13. Язык C++. Процедуры без параметров. Глобальные и локальные переменные.
14. Язык C++. Процедуры-операторы. Назначение, синтаксис, описание, активация (вызов). Формальные и фактические параметры. Параметры-значения. Параметры-переменные. Параметры-массивы. Глобальные и локальные переменные.
15. 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.
16. Среда программирования Visual 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)
17. 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.
 18. Статические и динамические программные объекты. Принципы их порождения, время их существования.
 19. Распределение памяти для глобальных переменных и локальных переменных процедур. Размеры сегментов памяти, выделяемых программе и статическим объектам. Размер динамической области памяти.
 20. Задание ссылочного типа переменных. Действия над ссылками. Пустая ссылка. Типизированные и нетипизированные указатели.
 21. Выделение и освобождение динамической памяти. Куча. Назначение. Примеры. Процедуры и функции для работы с динамической памятью. Примеры использования.
 22. Статические и динамические программные объекты. Принципы их порождения, возможность изменения в процессе работы программы объемов памяти, необходимых для их размещения, время их существования.
 23. Распределение памяти для глобальных переменных и локальных переменных процедур. Размеры сегментов памяти, выделяемых программе и статическим объектам. Размер динамической области памяти.
 24. Задание ссылочного типа переменных. Действия над ссылками. Пустая ссылка. Типизированные и нетипизированные указатели.
 25. Стеки. Стеки FIFO и LIFO. Дисциплины обслуживания стеков. Занесение элемента в стек. Выбор элемента из стека. Примеры использования стеков.
 26. Абстрактный тип списков. Виды списков. Дисциплина обработки списков. Примеры.
 27. Динамическая структура двунаправленного списка. Кольцевой список. Вставка, удаление и поиск элементов в двунаправленном и кольцевом списках. Примеры.
 28. Очереди. Набор базисных операций над очередями. Дисциплина обработки очередей. Примеры.
 29. Объектно-ориентированное программирование. Инкапсуляция. Наследование. Полиморфизм. Виртуальные методы. Объявление виртуальных методов. Конструкторы и деструкторы. Пример использования ООП.
 30. Классическая структура компилятора, ее составляющие.
 31. Этапы процесса компиляции.
 32. Лексический анализатор. Назначение. Прямой и непрямой лексический анализ. Механизм работы с таблицами. Хэширование. Хэш-функция.
 33. Синтаксический анализатор. Назначение, взаимодействие с семантическими процедурами. Дерево синтаксического анализа. Вопрос о неоднозначности грамматики.
 34. Инфиксная, постфиксная и префиксная форма записи предложений программы. Перевод арифметических и логических выражений в постфиксную запись.
 35. Формы промежуточного представления программы: ориентированный граф, тройки, четверки, префиксная или постфиксная запись, атрибутивное абстрактное дерево. Уровень промежуточного представления программы. Оптимизация кода.
 36. Основные концепции параллельного и распределенного программирования. Параллельная обработка данных.

Критерии формирования оценок по промежуточной аттестации

«Отлично» получают обучающиеся, которые свободно ориентируются в материале и отвечают без затруднений. Обучающийся способен к выполнению сложных

заданий, постановке целей и выборе путей их реализации. Работа выполнена полностью без ошибок, решено 100% задач;

«Хорошо» получают обучающиеся, которые относительно полно ориентируются в материале, отвечают без затруднений, допускают незначительное количество ошибок. Обучающийся способен к выполнению сложных заданий. Работа выполнена полностью, но имеются не более одной негрубой ошибки и одного недочета, не более трех недочетов. Допускаются незначительные неточности при решении задач, решено 70% задач;

«Удовлетворительно» получают обучающиеся, у которых недостаточно высок уровень владения материалом. В процессе ответа на экзамене допускаются ошибки и затруднения при изложении материала. 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;

«Неудовлетворительно» получают обучающиеся, которые допускают значительные ошибки. Обучающийся имеет лишь начальную степень ориентации в материале. В работе 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.

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 «Языки и методы программирования» является экзамен.

Общий балл текущего и рубежного контроля складывается из следующих составляющих (таблица 6).

Таблица 6

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 практических 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.
5	Первый этап (базовый уровень) –	не менее 36 б.	не менее 12 б.	не менее 12 б.	не менее 12 б.

	оценка «удовлетворительно»				
6	Второй этап (продвинутый уровень) – оценка «хорошо»	51-60 б.	менее 23 б.	менее 23 б.	менее 24 б.
7	Третий этап (высокий уровень) - оценка «отлично»	61-70 б.	не менее 23 б.	не менее 23 б.	не менее 24 б.

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

Критерии оценки качества освоения дисциплины

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.

Таблица 7

Результаты освоения учебной дисциплины, подлежащие проверке

Результаты обучения (компетенции)	Освоенные показатели оценки результатов обучения	Вид оценочного материала
<p>ПКС-1</p> <p>Способность проводить обследование организаций, выявлять информационные потребности пользователей, формировать требования к информационной системе</p>	<p>Знает: основные элементы и возможности языка программирования Си, основные алгоритмы обработки простых, структурированных и абстрактных данных на базе языка программирования Си, возможности интегрированной среды при написании программы, ее тестировании и отладке.</p> <p>Умеет: ставить задачу и разрабатывать алгоритм ее решения, анализировать методы решения задачи и обосновывать выбранный метод, обобщать данные при написании кода программы., работать с современными системами программирования, включая объектно-ориентированные, применять возможности интегрированной среды при написании программы, ее тестировании и отладке, применять основные алгоритмы обработки простых, структурированных и</p>	<p>Выполнение лабораторных работ</p> <p>Коллоквиум</p> <p>Тестирование</p> <p>Зачет</p> <p>(задания из раздела 5)</p>

	<p>абстрактных данных на базе языка программирования Си.</p> <p>Владеет: навыками разработки и отладки программ на одном из алгоритмических процедурных языков программирования высокого уровня</p>	
<p>ПКС-2</p> <p>Способность разрабатывать и адаптировать прикладное программное обеспечение</p>	<p>Знает: этапы трансляции программы.</p> <p>Умеет: 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.</p> <p>Владеет: 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.</p>	<p>Выполнение лабораторных работ</p> <p>Коллоквиум</p> <p>Тестирование</p> <p>Зачет</p> <p>(задания из раздела 5)</p>
<p>ПКС-6</p> <p>Способность принимать участие в организации ИТ-инфраструктуры и управлении информационной безопасностью</p>	<p>Обеспечивает Знать инженерно-техническую поддержку организации ИТ-инфраструктуры предприятия</p> <p>Использует данным технологии интеграции требований организации ИС с существующей ИТ-инфраструктурой предприятия заказчика технологии</p> <p>Обеспечивает управление доступом к данным с учетом требований организации ИТ-инфраструктуры</p>	<p>Выполнение лабораторных работ</p> <p>Коллоквиум</p> <p>Тестирование</p> <p>Зачет</p> <p>(задания из раздела 5)</p>

7. Учебно-методическое обеспечение дисциплины

7.1. 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»

7.2. 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

2. Павловская Т.А. C/C++. Программирование на языке высокого уровня. Питер, 2013. — 464 с. — (Стандарт 3-го поколения). — Режим доступа: <http://www.iprbookshop.ru/68524.html>.— ЭБС «IPRbooks»

3. Липпман Стенли Б., Лажойе Жози, Му Барбара Э. Язык программирования C++. Базовый курс. М.: Вильямс, 2014. – 1120 с. – 5-е изд.

4. 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»

7.3. Интернет-ресурсы

1. <http://iprbookshop.ru>

2. ГАРАНТ.РУ. Информационно-правовой портал [Электронный ресурс]. – Режим доступа: <http://www.garant.ru>, свободный. – Загл. с экрана;

3. КонсультантПлюс. Официальный сайт компании «Консультант-Плюс» [Электронный ресурс]. – Режим доступа: <http://www.consultant.ru>, свободный.– Загл. с экрана;

7.4. 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)

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

Методические указания к практическим 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 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 является формой итогового контроля знаний и умений обучающихся по данной дисциплине, полученных на лекциях, лабораторных 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 выносится 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

Лекции по дисциплине проводятся в аудитории, оснащенной видеопроектором.

– Лабораторные занятия проводятся в компьютерных классах института, компьютеры которых оснащены необходимым лицензионным программным обеспечением – Visual Studio.

Студенты имеют доступ через сеть «Интернет» к единому образовательному portalу, где в открытом доступе имеются ресурсы учебно-методической литературы, являющиеся разработками ведущих ВУЗов России.

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:

1. Альтернативной версией официального сайта в сети «Интернет» для слабовидящих.
2. Присутствие ассистента, оказывающего обучающемуся необходимую помощь.
3. Для инвалидов и лиц с ограниченными возможностями здоровья по слуху – дублирование вслух справочной информации о расписании учебных занятий; обеспечение надлежащими звуковыми средствами воспроизведения информации.
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, объекту питания, туалетные и другие помещения университета, а также пребывания в указанных помещениях (наличие расширенных дверных проемов, поручней и других приспособлений).

9.ЛИСТ ИЗМЕНЕНИЙ (ДОПОЛНЕНИЙ)

в рабочую программу по дисциплине «Программирование на языке высокого уровня»
по направлению подготовки 09.03.03 Прикладная информатика, Профиль
Корпоративные информационные системы на _____ учебный год

№п/п	Элемент (пункт) РПД	Перечень вносимых изменений (дополнений)	Примечание

Обсуждена и рекомендована на заседании кафедры компьютерных технологий и
информационной безопасности протокол № _____ от " ____ " _____ 20__ г.

Заведующий кафедрой _____ / _____ /