

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Чувашский государственный университет имени И.Н.Ульянова»
Факультет информатики и вычислительной техники
Кафедра математического и аппаратного обеспечения информационных систем

«УТВЕРЖДАЮ»
Проректор по учебной работе

И.Е. Поверинов

«31» августа 2017 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
«ПРОГРАММИРОВАНИЕ»

Направление подготовки (специальность) 09.03.01 Информатика и вычислительная техника
Квалификация (степень) выпускника – Бакалавр
Профиль (направленность) Вычислительные машины, комплексы, системы и сети
Академический бакалавриат

Чебоксары – 2017

Рабочая программа основана на требованиях Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.01 Информатика и вычислительная техника, утвержденного приказом Министерства образования и науки Российской Федерации №5 от 12.01.2016 г.

СОСТАВИТЕЛЬ:

Доцент



Л.А. Ильина

ОБСУЖДЕНО:

на заседании кафедры математического и аппаратного обеспечения информационных систем 30 августа 2017 г., протокол № 1

заведующий кафедрой



Д.В. Ильин

СОГЛАСОВАНО:

Методическая комиссия факультета информатики и вычислительной техники «30» августа 2017 г., протокол № 1.

Декан факультета



А.В. Щипцова

Директор научной библиотеки



Н.Д. Никитина

Начальник управления информатизации



И.П. Пивоваров

Начальник учебно-методического управления



В.И. Маколов

Оглавление

1. Цель и задачи обучения по дисциплине	4
2. Место дисциплины в структуре основной образовательной программы (ООП)	4
3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ООП	4
4. Структура и содержание дисциплины	5
4.1. Содержание дисциплины	5
4.2. Объем дисциплины, виды учебной работы обучающихся по очной форме обучения	6
4.3. Содержание разделов дисциплины	6
4.3.1. Лекции	6
4.3.2. Лабораторные работы	8
4.3.3. Вопросы для самостоятельной работы студента в соответствии с содержанием разделов дисциплины	8
5. Образовательные технологии	8
6. Формы аттестации и оценочные материалы	9
6.1. Вопросы и задачи к зачету	9
6.2. Вопросы и задачи к экзаменам	10
6.3. Выполнение и примерная тематика курсового проекта	14
7. Учебно-методическое и информационное обеспечение дисциплины	16
7.1. Рекомендуемая основная литература	16
7.2. Рекомендуемая дополнительная литература	16
7.3. Программное обеспечение, профессиональные базы данных, информационно-справочные системы.	16
8. Материально-техническое обеспечение дисциплины	17
9. Средства адаптации преподавания дисциплины к потребностям лиц с ограниченными возможностями	17
10. Методические рекомендации по освоению дисциплины	18

1. Цель и задачи обучения по дисциплине

Цель дисциплины изучение принципов программирования, средств для разработки программ, методов отладки и решения задач на ЭВМ в различных режимах; основ объектно-ориентированного подхода к программированию

Задачи дисциплины:

- Освоение методик использования программных средств для решения практических задач.
- Изучение программных средств, языков и систем программирования;
- Освоение технологий разработки алгоритмов и программ;
- Овладение методами отладки и решения задач на ЭВМ в различных режимах;
- Применение современных инструментальных средств при разработке программного обеспечения.

2. Место дисциплины в структуре основной образовательной программы (ООП)

Дисциплина «Программирование» является обязательной дисциплиной базовой части учебного образовательной программы по направлению 09.03.01 «Информатика и вычислительная техника».

Изучение дисциплины «Программирование» основывается на базе знаний, умений и владений, полученных обучающимися на предыдущем уровне образования.

Требования к входным знаниям, умениям и владениям обучающимися:

Дисциплина изучается с 1 семестра, поэтому знания, умения и навыки еще не сформированы.

Дисциплина является базовым теоретическим и практическим основанием для следующих дисциплин и практик: «Методы вычислений», «Базы данных», «Цифровая обработки сигналов», «Конструкторское технологическое обеспечение производства ЭВМ», «Компьютерная графика», прохождения практик, выполнения выпускной квалификационной работы.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ООП

Процесс обучения по дисциплине направлен на формирование следующих компетенций:

Компетенция по ФГОС	Основные показатели обучения ()
способность инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем (ОПК-1)	<p>знать: методы и этапы инсталляции программного обеспечения (31),</p> <p>уметь: инсталлировать программное обеспечение (У1),</p> <p>владеть навыками: инсталляции и использования интегрированных средств разработки для создания программных продуктов (Н1)</p>
способность осваивать методики использования программных средств для решения практических задач (ОПК-2)	<p>знать: синтаксис и семантику языка программирования высокого уровня (32), основные структуры данных и способы их реализации (33),</p> <p>уметь: ставить задачу и разрабатывать алгоритм ее решения (У2),</p> <p>владеть навыками: проектирования алгоритмов и реализации их на языках программирования (Н2)</p>

<p>способность разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования (ПК-2)</p>	<p>знать: этапы разработки программных средств и принципы построения алгоритмов (З4);</p> <p>уметь: проектировать и реализовывать алгоритмы с помощью современных средств программирования (У3);</p> <p>владеть навыками: отладки и тестирования алгоритмов (Н3)</p>
---	---

4. Структура и содержание дисциплины

Образовательная деятельность по дисциплине проводится:

- в форме контактной работы обучающихся с педагогическими работниками организации и (или) лицами, привлекаемыми организацией к реализации образовательных программ на иных условиях (далее – контактная работа);
- в форме самостоятельной работы.

Контактная работа включает в себя занятия лекционного типа, занятия семинарского типа (семинары, практические занятия, лабораторные работы, практикумы), групповые и (или) индивидуальные консультации, в том числе в электронной информационно-образовательной среде.

Обозначения:

Л – лекции, л/р – лабораторные работы, КСР – контроль самостоятельной работы, СРС – самостоятельная работа студента, ИФР – интерактивная форма работы, К – контроль.

4.1. Содержание дисциплины

Содержание	Формируемые компетенции	Формируемые ЗУН
Раздел 1. Технология разработки алгоритмов и приложений	ОПК-1, ПК-2	31, 34, У1, Н1, Н3
1.1. Организация взаимодействия пользователя с ЭВМ.		
1.2. Общая характеристика языков программирования, области их применения.		
1.3. Основные этапы разработки приложений. Способы описания алгоритмов		
1.4. Тестирование и отладка		
Раздел 2. Программирование на языке высокого уровня	ОПК-1, ОПК-2, ПК-2	32, 33, У2, У3, Н1, Н2, Н3
2.1. Элементы языка. Структура программы. Операторы.		
2.2. Ввод-вывод данных		
2.3. Циклы		
2.4. Массивы		
2.5. Структурные типы и структуры.		
2.6. Обработка текстовой информации		
2.7. Файлы		
2.8. Тип данных класс. Объект.		
2.9. Инкапсуляция		
2.10. Наследование		
2.11. Полиморфизм		
2.12. Виртуальные методы		
2.13. Конструкторы и деструкторы		
2.14. Динамические структуры данных.		
2.15. Сортировка и поиск с использованием динамических структур данных		
Курсовой проект	ОПК-1, ОПК-2, ПК-2	31, 32, 33, 34, У1, У2, У3, Н1, Н2, Н3
Экзамены, Зачет	ОПК-1, ОПК-2, ПК-2	31, 32, 33, 34 У1, У2, У3 Н1, Н2, Н3

4.2. Объем дисциплины, виды учебной работы обучающихся по очной форме обучения

Содержание	Всего, час	Контактная работа, час				СРС, час	ИФР, час	К, час
		Л	л/р	п/р	КСР			
Раздел 1. Технология разработки алгоритмов и приложений								
1.1. Организация взаимодействия пользователя с ЭВМ.	4	2	2				2	
1.2. Общая характеристика языков программирования, области их применения.	6	2				4	2	
1.3. Основные этапы разработки приложений. Способы описания алгоритмов	6	2	2			2	2	
1.4. Тестирование и отладка	6	2	2			2	2	
Раздел 2. Программирование на языке высокого уровня								
2.1. Элементы языка. Структура программы. Операторы.	6	2	2			2	4	
2.2. Ввод-вывод данных	12	2	6			4	4	
2.3. Циклы	12	2	8			2	8	
2.4. Массивы	12	2	8			2	8	
2.5. Функции	14	4	8			2	8	
2.6. Указатели	10	2	6			2	6	
2.7. Структурные типы и структуры.	10	2	6			2	6	
2.8. Обработка текстовой информации	14	4	8			2	8	
2.9. Файлы	14	4	8			2	8	
2.10. Динамические структуры данных.	26	8	16			2	12	
2.11. Тип данных класс. Объект.	12	4	8				8	
2.12. Инкапсуляция	16	4	10			2	10	
2.13. Наследование	18	6	10			2	10	
2.14. Полиморфизм	22	6	10			6	10	
2.15. Виртуальные методы	18	4	8			6	10	
Зачет	2					2		
Курсовой проект	12				2	10		
Экзамены 1, 2 семестр	72							72
Итого	324 9 з.е.	64	128		2	58	128	72

4.3. Содержание разделов дисциплины

4.3.1. Лекции

Раздел 1. Технология разработки алгоритмов и приложений

1.1. Организация взаимодействия пользователя с ЭВМ.

Назначение, состав и структура программного обеспечения. Организация взаимодействия пользователя с ЭВМ.

1.2. Общая характеристика языков программирования, области их применения.

Общая характеристика языков программирования, области их применения. Компиляторы и интерпретаторы.

1.3. Основные этапы разработки приложений. Способы описания алгоритмов

Технология разработки алгоритмов и приложений. Основные этапы разработки приложений. Определение алгоритма. Свойства алгоритма. Способы описания алгоритмов: словесный, схемный, с помощью псевдокода или языка программирования. Методы разработки алгоритмов и программ: нисходящее, восходящее. Модульное представление программ.

1.4. Тестирование и отладка

Методы тестирования. Типы ошибок. Способы и средства обнаружения и локализации синтаксических и логических ошибок. Организация отладки и тестирования приложений.

Раздел 2. Программирование на языке высокого уровня

2.1. Элементы языка. Структура программы. Операторы.

Алфавит, лексемы, идентификаторы, ключевые слова. Переменные и константы. Знаки операций. Приоритеты операций. Условный оператор и переключатель. Структура программы. Определение констант и типов данных. Приведение типов и функции преобразования типов. Операторы.

2.2. Ввод-вывод данных

Форматированный ввод и вывод данных. Поточковый ввод и вывод данных.

2.3. Циклы

Цикл с предусловием, цикл с постусловием и цикл с параметром. Операторы break и continue.

2.4. Массивы

Одномерные и многомерные массивы. Инициализация и ввод массивов. Стандартные алгоритмы обработки массивов. Выделение памяти. Динамические массивы.

2.5. Функции

Объявление функции. Механизмы передачи параметров. Локальные и глобальные параметры. Область видимости и время жизни переменной. Рекурсивные функции. Передача параметров. Функции с переменным количеством параметров. Перегрузка и шаблоны функций.

2.6. Указатели

Адреса, указатели и ссылки. Указатели в параметрах функций. Указатели на функции, массивы указателей на функции.

2.7. Структурные типы и структуры.

Объявление структурного типа, структуры, указателя на структуру. Доступ к компонентам структур. Массивы структур. Структуры и функции.

2.8. Обработка текстовой информации

Способы представления текстов. Символы и строки. Встроенные подпрограммы обработки строк.

2.9. Файлы

Текстовые, типизированные и двоичные файлы. Прямой и последовательный доступ.

2.10. Динамические структуры данных.

Организация динамических структур данных. Списки, стеки, очереди, деревья.

2.11. Тип данных класс. Объект.

Концепции объектно-ориентированного программирования. Определение класса и его членов. Конструкторы и деструкторы. Объекты. Массивы объектов.

2.12. Инкапсуляция

Закрытые, открытые и защищенные члены класса. Обеспечение доступа. Дружественные функции.

2.13. Наследование

Организация наследования классов. Иерархия и множественное наследование. Конструкторы, деструкторы и наследование.

2.14. Полиморфизм

Перегрузка методов в классе. Конструкторы копий. Перегрузка операторов. Перегрузка ввода-вывода.

2.15. Виртуальные методы

Виртуальные классы и виртуальные методы.

4.3.2. Лабораторные работы

№	Тема	Количество часов
Лабораторная работа №1.	1.1. Организация взаимодействия пользователя с ЭВМ.	2
Лабораторная работа №3.	1.3. Основные этапы разработки приложений. Способы описания алгоритмов	2
Лабораторная работа №4.	1.4. Тестирование и отладка	2
Лабораторная работа №5.	2.1. Элементы языка. Структура программы. Операторы.	2
Лабораторная работа №6.	2.2. Ввод-вывод данных	6
Лабораторная работа №7.	2.3. Циклы	8
Лабораторная работа №8.	2.4. Массивы	8
Лабораторная работа №9.	2.5. Функции	8
Лабораторная работа №10.	2.6. Указатели	6
Лабораторная работа №11.	2.7. Структурные типы и структуры.	6
Лабораторная работа №12.	2.8. Обработка текстовой информации	8
Лабораторная работа №13.	2.9. Файлы	8
Лабораторная работа №14.	2.10. Динамические структуры данных.	6
Лабораторная работа №15.	2.11. Тип данных класс. Объект.	8
Лабораторная работа №16.	2.12. Инкапсуляция	10
Лабораторная работа №17.	2.13. Наследование	10
Лабораторная работа №18.	2.14. Полиморфизм	10
Лабораторная работа №19.	2.15. Виртуальные методы	8
Итого		128

4.3.3. Вопросы для самостоятельной работы студента в соответствии с содержанием разделов дисциплины

Тема	Вопрос
1.2. Общая характеристика языков программирования, области их применения.	Обзор систем программирования
1.3. Основные этапы разработки приложений. Способы описания алгоритмов	Единая система программной документации: содержание, вид, форма.
1.4. Тестирование и отладка	Сравнительный анализ компиляторов и интерпретаторов
2.1. Элементы языка. Структура программы. Операторы.	Сравнительный анализ языков С и С++
2.2. Ввод-вывод данных	Манипуляторы потокового вывода
2.3. Циклы	Вложение циклов
2.4. Массивы	Алгоритмы сортировки массивов
2.5. Функции	Косвенные рекурсии
2.6. Указатели	Использование указателей и ссылок в функциях
2.7. Структурные типы и структуры.	Вложение структур
2.8. Обработка текстовой информации	Библиотечные функции обработки строк
2.9. Файлы	Позиционирование в файлах
2.10. Динамические структуры данных.	Сортировка и поиск с использованием динамических структур данных
2.12. Инкапсуляция	Скрытие данных в классах
2.13. Наследование	Передача параметров при наследовании
2.14. Полиморфизм	Пользовательские манипуляторы
2.15. Виртуальные методы	Чисто виртуальные функции

5. Образовательные технологии

В соответствии со структурой образовательного процесса по дисциплине

применяются следующие технологии:

- диагностики;
- целеполагания;
- управления процессом освоения учебной информации;
- применения знаний на практике, поиска новой учебной информации;
- организации совместной и самостоятельной деятельности обучающихся (учебно-познавательной, научно-исследовательской, частично-поисковой, репродуктивной, творческой и пр.);
- контроля качества и оценивания результатов образовательной деятельности (технология оценивания качества знаний, рейтинговая технология оценки знаний и др.)

В соответствии с требованиями ФГОС ВО для реализации компетентного подхода при обучении дисциплине предусмотрено широкое использование в учебном процессе активных и интерактивных методов проведения занятий:

При обучении дисциплине применяются следующие формы занятий:

- лекции, направленные на получение новых и углубление научно-теоретических знаний, в том числе вводная лекция, информационная лекция, обзорная лекция, лекция-консультация, проблемная лекция, лекции-дискуссии, лекции-беседы и др.;
- лабораторные занятия, проводимые под руководством преподавателя в учебной лаборатории с использованием компьютеров и учебного оборудования, направленные на закрепление и получение новых умений и навыков, применение знаний и умений, полученных на теоретических занятиях, при решении практических задач и др.

Все занятия обеспечены мультимедийными средствами (SMART доски, проекторы, экраны) для повышения качества восприятия изучаемого материала. В образовательном процессе широко используются информационно-коммуникационные технологии.

Самостоятельная работа студентов – это планируемая работа студентов, выполняемая по заданию при методическом руководстве преподавателя, но без его непосредственного участия. Формы самостоятельной работы студентов определяются содержанием учебной дисциплины, степенью подготовленности студентов. Они могут иметь учебный или учебно-исследовательский характер: анализ литературы по теме, подготовка к лабораторным работам, подготовка реферативных сообщений, разработка проекта и др.

Формами контроля самостоятельной работы выступают оценивание проверка отчётов по результатам выполненных заданий и лабораторных работ, курсового проекта. Результаты самостоятельной работы учитываются при оценке знаний на экзамене и зачёте.

6. Формы аттестации и оценочные материалы

Текущий контроль предназначен для проверки хода и качества усвоения учебного материала, стимулирования учебной работы обучающихся и совершенствования методики проведения занятий. Он проводится в ходе всех видов занятий в форме, избранной преподавателем.

Промежуточная аттестация имеет целью определить степень достижения учебных целей по учебной дисциплине и проводится в форме экзаменов и зачета, защиты курсового проекта. Принимается экзамен и зачет преподавателями, читающими лекции по данной учебной дисциплине в соответствии с перечнем основных вопросов, выносимых для контроля знаний обучающихся:

6.1. Вопросы и задачи к зачету

1. Общая характеристика языков программирования, области их применения.
2. Компиляторы и интерпретаторы.
3. Основные этапы разработки приложений.
4. Определение алгоритма. Свойства алгоритма.

5. Способы описания алгоритмов.
6. Методы разработки алгоритмов и программ: нисходящее, восходящее.
7. Модульное представление программ.
8. Методы тестирования. Типы ошибок.
9. Способы и средства обнаружения и локализации синтаксических и логических ошибок.
10. Перечислите способы, с помощью которых можно организовать ветвления в C++. Приведите примеры.
11. Алгоритмы циклической структуры и способы их организации. Прерывания циклов. Приведите примеры.
12. Особенности использования вложенных циклов.
13. Операторы break и continue для чего применяются. Примеры.
14. Особенности использования массивов в C++. Одномерные и многомерные массивы.
15. Инициализация массивов. Примеры использования для решения задач.
16. Функции, определяемые пользователем. Параметры. Примеры.
17. Подпрограммы в C++. Особенности их организации.
18. Указатели. Операции над указателями.
19. Указатели и массивы.
20. Массивы динамической памяти.
21. Работа с текстовой информацией.
22. Массивы в параметрах функций.
23. Строки в параметрах функций.
24. Указатели и функции.
25. Функции с переменным количеством параметров.
26. Производные типы. Определение. Выделение памяти.
27. Массивы и структуры как элементы структур.
28. Массивы структур.
29. Указатели на структуры.
30. Форматированный ввод-вывод в C++.
31. Файловый ввод-вывод.

Оценивание результатов зачета

Зачет проводится по окончании занятий по дисциплине до начала экзаменационной сессии в период недели контроля самостоятельной работы.

Билет для проведения промежуточной аттестации в форме зачета включают вопросы и задачи для проверки сформированности знаний, умений и навыков.

Оценка «зачтено» проставляется студенту, выполнившему и защитившему в полном объеме лабораторные работы в течение семестра, чей уровень знаний, умений и навыков соответствует уровню оценок «отлично», «хорошо» или «удовлетворительно» (п.7.2). Оценка «не зачтено» проставляется студенту, не выполнившему и (или) не защитившему в полном объеме лабораторные работы в течение семестра, либо чей уровень знаний, умений и навыков соответствует уровню оценки «неудовлетворительно».

6.2. Вопросы и задачи к экзаменам

1 семестр

1. Общая характеристика языков программирования, области их применения.
2. Компиляторы и интерпретаторы.
3. Основные этапы разработки приложений.
4. Определение алгоритма. Свойства алгоритма.
5. Способы описания алгоритмов.
6. Методы разработки алгоритмов и программ: нисходящее, восходящее.
7. Модульное представление программ.

8. Методы тестирования. Типы ошибок.
9. Способы и средства обнаружения и локализации синтаксических и логических ошибок.
10. Организация ветвлений в C++.
11. Организация циклов, прерывания циклов.
12. Особенности использования вложенных циклов.
13. Особенности использования одномерных массивов в C++.
14. Особенности использования многомерных массивов.
15. Инициализация массивов. Примеры использования для решения задач.
16. Функции, определяемые пользователем.
17. Подпрограммы в C++. Особенности их организации.
18. Указатели. Операции над указателями.
19. Указатели и отношения.
20. Указатели и массивы.
21. Массивы динамической памяти.
22. Строки в C++.
23. Массивы в параметрах функций.
24. Строки в параметрах функций.
25. Указатели и функции.
26. Функции с переменным количеством параметров.
27. Производные типы. Определение. Выделение памяти.
28. Массивы и структуры как элементы структур.
29. Массивы структур.
30. Указатели на структуры.
31. Форматированный ввод-вывод в C++.
32. Файловый ввод-вывод.

Примерные задачи:

Циклы

1. Дано натуральное число N . Вычислить: $S = 1 - \frac{1}{2} + \frac{1}{4} - \frac{1}{8} + \dots + (-1)^n \cdot \frac{1}{2^n}$.

2. Дано натуральное число N . Вычислить:

$$S = \frac{1}{\sin 1} + \frac{1}{\sin 1 + \sin 2} + \dots + \frac{1}{\sin 1 + \sin 2 + \dots + \sin N}.$$

3. Дано натуральное число N . Вычислить произведение первых N сомножителей

$$P = \frac{2}{3} \cdot \frac{4}{5} \cdot \frac{6}{7} \times \dots \times \frac{2N}{2N+1}.$$

4. Дано натуральное число N . Вычислить:

$$\frac{\cos 1}{\sin 1} \times \frac{\cos 1 + \cos 2}{\sin 1 + \sin 2} \times \dots \times \frac{\cos 1 + \cos 2 + \dots + \cos N}{\sin 1 + \sin 2 + \dots + \sin N}.$$

5. Дано действительное число x . Вычислить: $x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \frac{x^{11}}{11!} + \frac{x^{13}}{13!}$.

Одномерные массивы

1. Дан массив натуральных чисел. Найти сумму элементов, кратных данному числу K .

2. В целочисленной последовательности есть нулевые элементы. Создать массив из номеров этих элементов.

3. Дана последовательность чисел a_1, a_2, \dots, a_n . Выяснить, какое число встречается раньше положительное или отрицательное.

4. Дана последовательность действительных чисел a_1, a_2, \dots, a_n . Является ли она возрастающей.

5. Дана последовательность натуральных чисел a_1, a_2, \dots, a_n . Создать массив из четных чисел этой последовательности. Если таких чисел нет, вывести соответствующее

сообщение.

Двумерные массивы

1. Задана квадратная матрица. Получить транспонированную матрицу (столбцы и строки меняются местами).
2. Квадратная матрица, симметричная относительно главной диагонали, задана верхним треугольником в виде одномерного массива. Восстановить исходную матрицу и напечатать ее.
3. Задана матрица порядка n и число k . Разделить элементы k -й строки на диагональный элемент, расположенный в этой строке.
4. Для целочисленной квадратной матрицы найти число элементов, кратных k , и наибольший из полученных результатов.
5. Для целочисленной квадратной матрицы найти число элементов, кратных k , и их произведение.

Функции

1. Треугольник задан координатами своих вершин. Составить программу вычисления его площади.
2. Составить программу нахождения наибольшего общего делителя и наименьшего общего кратного двух натуральных чисел.
3. Составить программу нахождения наибольшего общего делителя четырех натуральных чисел.
4. Составить программу нахождения наименьшего общего кратного трех натуральных чисел.
5. Написать программу нахождения суммы большего и меньшего из трех чисел.

Обработка текстовой информации

1. Дана строка, заканчивающаяся точкой. Подсчитать, сколько в ней слов.
2. Дана строка, содержащая английский текст. Найти количество слов, начинающихся с буквы b .
3. Дана строка. Подсчитать в ней количество вхождений букв f, k, t .
4. Дана строка. Определить, сколько в ней символов $*$; $;$.
5. Дана строка, содержащая текст. Найти длину самого короткого слова и самого длинного слова.

2 семестр

1. Основные концепции объектно-ориентированного программирования.

Примеры.

2. Объявление класса. Типы членов класса.
3. Определение функций-членов класса.
4. Конструкторы и деструкторы.
5. Конструкторы с параметрами.
6. Указатели на объекты.
7. Встраиваемые функции, их преимущества и недостатки.
8. Использование встраиваемых функций в объявлении объектов.
9. Присваивание объектов.
10. Передача объектов функциям. Особенности.
11. Объекты в качестве возвращаемого функцией значения. Особенности.
12. Дружественные функции.
13. Предварительное объявление класса.
14. Массивы объектов. Объявление и инициализация.
15. Указатель `this`.
16. Способы выделения памяти в C++. Преимущества и недостатки каждого.
17. Ссылки. Способы их использования. Преимущества и недостатки.
18. Перегрузка функций.
19. Перегрузка конструктора

20. Конструктор копий.
21. Аргументы по умолчанию и их использование при перегрузке функций.
22. Перегрузка операторов.
23. Дружественные оператор-функции.
24. Перегрузка оператора присваивания.
25. Перегрузка оператора индекса массива.
26. Наследование.
27. Защищенные члены класса.
28. Конструкторы, деструкторы и наследование.
29. Пользовательские функции вывода.
30. Пользовательские функции ввода.
31. Поточковый ввод-вывод
32. Шаблоны функций.
33. Виртуальные функции.

Темы задач:

Определение классов. Наследование. Виртуальные функции

1. Создать класс «Треугольник», заданный координатами вершин. Определить производные классы «Равносторонний» и «Прямоугольный» треугольники. Предусмотреть виртуальную функцию для вычисления площади описанной около треугольника окружности, сравнить площади окружностей, описанных около равностороннего и прямоугольного треугольников, координаты которых введены с клавиатуры.

2. Создать класс «Окружность», заданный координатами центра и точкой на окружности. Определить производный класс «Описанная окружность». Предусмотреть виртуальную функцию для вычисления периметра равностороннего треугольника, вписанного в окружность. Сравнить периметры 3 равносторонних треугольников, вписанных в окружности, данные о которых вводятся с клавиатуры.

3. Создать класс «Окружность», заданный координатами центра и точкой на окружности. Определить производный класс «Описанная окружность». Предусмотреть виртуальную функцию для вычисления площади равностороннего треугольника, вписанного в окружность. Сравнить площади 2 равносторонних треугольников, вписанных в окружности, данные о которых вводятся с клавиатуры.

4. Создать класс «Шар», заданный координатами центра и точкой на сфере. Определить производный класс «Окружность». Предусмотреть виртуальную функцию для вычисления площади круга, сравнить площади 2 кругов, данные о которых введены с клавиатуры.

5. Создать класс «Шар», заданный радиусом. Определить производный класс «Окружность». Предусмотреть виртуальную функцию для вычисления площади круга и длины окружности, сравнить площади и длины 2 окружностей, радиусы которых введены с клавиатуры.

Потоковый ввод-вывод. Текстовые и бинарные потоки. Перегрузка ввода-вывода

1. В одном файле хранятся сведения о книгах: код, автор, название, во 2 файле сведения об издании: код, издательство, год издания. Вывести на экран информацию о книгах автора, введенного с клавиатуры, изданных в 2013 году, введенного с клавиатуры (ФИО, должность, зарплата за год), выбрать сотрудников, получивших зарплату за год меньше средней по отделу.

2. В одном файле хранятся сведения о сотрудниках: ФИО, табельный номер, должность, отдел, во 2 файле сведения о работе за год: табельный номер, заработная плата по месяцам. Вывести на экран информацию о сотрудниках отдела, введенного с клавиатуры (ФИО, должность, зарплата за год), выбрать сотрудников, получивших зарплату за год меньше средней по отделу.

3. В одном файле хранятся сведения о товарах в магазине: код товара, название, цена, во 2 файле сведения о производстве: код товара, производитель, дата производства, срок годности. Вывести на экран информацию о товарах произведенных после 20.06.2012 (название, цена, производитель, дата производства), которые стоят менее 300 рублей, у которых истек срок годности.

4. В одном файле хранятся сведения о сотрудниках: ФИО, табельный номер, должность, отдел, во 2 файле сведения о работе за год: табельный номер, заработная плата по месяцам. Вывести на экран информацию о сотрудниках отдела, введенного с клавиатуры (ФИО, должность, зарплата за год), отсортированную по фамилиям в алфавитном порядке.

5. В одном файле хранятся сведения об учениках: ФИО, класс, во 2 файле сведения о результатах сдачи экзаменов: ФИО, название предмета, ФИО преподавателя, оценка. Вывести на экран фамилии, класс и оценки учеников, сдавших математику, но не сдавших русский язык.

Оценивание результатов экзамена

Экзаменационный билет для проведения промежуточной аттестации включают вопросы и задачи для проверки сформированности знаний, умений и навыков.

Общими критериями, определяющими оценку знаний, умений и навыков на экзамене, являются:

- для оценки «отлично» - наличие глубоких и исчерпывающих знаний в объеме пройденного программного материала правильные и уверенные действия по применению полученных знаний на практике, грамотное и логически стройное изложение материала при ответе, знание дополнительно рекомендованной литературы;

- для оценки «хорошо» - наличие твердых и достаточно полных знаний программного материала, незначительные ошибки при освещении заданных вопросов, правильны действия по применению знаний на практике, четкое изложение материала;

- для оценки «удовлетворительно» - наличие твердых знаний пройденного материала, изложение ответов с ошибками, уверенно исправляемыми после дополнительных вопросов, необходимость наводящих вопросов, правильные действия по применению знаний на практике;

- для оценки «неудовлетворительно» - наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неумение применять знания на практике, неуверенность и неточность ответов на дополнительные и наводящие вопросы.

6.3. Выполнение и примерная тематика курсового проекта

Курсовой проект выполняется в процессе изучения дисциплины. Общее руководство и контроль за ходом выполнения курсового проекта осуществляет преподаватель соответствующей дисциплины. Курсовой проект выполняется в соответствии с методическими указаниями для обучающихся.

Основными функциями руководителя курсового проекта являются:

- определение и формулирование темы курсового проекта совместно с обучающимся на основе примерной тематики или по предложенной обучающимся теме в рамках содержания дисциплины;

- консультирование по вопросам содержания и последовательности выполнения курсового проекта;

- оказание помощи студенту в подборе необходимой литературы;

- контроль хода выполнения курсового проекта.

Примерная тематика курсовых проектов:

1. Проектирование и реализация класса «Сотрудники института»
2. Проектирование и реализация класса «Объекты-векторы в трехмерном пространстве»
3. Проектирование и реализация класса «Записная книжка»

4. Проектирование и реализация класса «Массивы строк фиксированной длины»
5. Проектирование и реализация класса «Сведения о личной коллекции книголюбца»
6. Проектирование и реализация класса «Почтовый адрес организации»
7. Проектирование и реализация класса «Сведения о сдаче сессии»
8. Использование дополненной реальности в образовании
9. Проектирование и реализация класса «Обыкновенные дроби»
10. Проектирование и реализация класса «Расписание поездов»
11. Проектирование и реализация класса «Многочлены от одной переменной»
12. Проектирование и реализация класса «Работа со строками»
13. Проектирование и реализация класса «Пациенты глазной клиники»
14. Проектирование и реализация класса «Работа аэропорта»
15. Проектирование и реализация класса «Об игроках хоккейных команд»
16. Проектирование и реализация класса «Рабочие цеха»
17. Проектирование и реализация класса «Сдача студентами сессии»
18. Проектирование и реализация класса «Телефонные абоненты»
19. Проектирование и реализация класса «Одномерный массив строк»
20. Проектирование и реализация класса «Матрица»
21. Проектирование и реализация класса «Выбор дисциплины студентами»
22. Проектирование и реализация класса «Одномерные массивы»
23. Проектирование и реализация класса «Одномерные массивы целых чисел»
24. Проектирование и реализация класса «Домашняя библиотека»
25. Проектирование и реализация класса «Студенческая группа»
26. Проектирование и реализация класса «Вещественная матрица»
27. Проектирование и реализация класса «Публичная библиотека»

Оценивание курсового проекта

Курсовой проект выполняется в процессе изучения дисциплины. Общее руководство и контроль за ходом выполнения курсового проекта осуществляет преподаватель соответствующей дисциплины. Курсовой проект выполняется в соответствии с методическими указаниями для обучающихся.

Оценивание курсового проекта осуществляется в соответствии с полнотой и качеством выполнения задания на курсовой проект, качеством защиты проекта (ответы на вопросы, презентация и др.). Оценка курсового проекта отражает уровень сформированности соответствующих компетенций:

– «отлично» - проект выполнен в соответствии с утвержденным планом и заданием, полностью раскрыто содержание каждого вопроса; студентом сформулированы собственные аргументированные выводы по теме работы; оформление работы соответствует предъявляемым требованиям; при защите работы обучающийся демонстрирует свободное владение материалом и верно отвечает на поставленные вопросы;

– «хорошо» - проект выполнен в соответствии с утвержденным планом и заданием; полностью раскрыто содержание каждого вопроса; имеются незначительные замечания к оформлению работы; при защите работы обучающийся демонстрирует владение материалом, но отвечает на ряд поставленных вопросов не в достаточно полном объеме;

– «удовлетворительно» - проект выполнен в соответствии с утвержденным планом и заданием, но не полностью раскрыто содержание каждого вопроса; обучающимся не сделаны собственные выводы по теме работы; допущены существенные недостатки в оформлении работы; при защите работы обучающийся демонстрирует владение материалом, но отвечает не на все поставленные вопросы, либо не в достаточно полном объеме;

– «неудовлетворительно» - если проект не выполнена в соответствии с утвержденным планом и заданием, не раскрыто содержание каждого вопроса; обучающимся не сделаны выводы по теме работы, имеются существенные недостатки в оформлении работы; при защите работы обучающийся не демонстрирует владение материалом, не отвечает на поставленные вопросы.

В случае оценивания проекта на «неудовлетворительно» работа направляется на дальнейшую доработку.

7. Учебно-методическое и информационное обеспечение дисциплины

Электронный каталог и электронные информационные ресурсы, предоставляемые научной библиотекой ФГБОУ ВО «Чувашский государственный университет имени И.Н. Ульянова» доступны по ссылке <http://library.chuvsu.ru/>

7.1. Рекомендуемая основная литература

№	Название
1.	Васильев В.Н. Основы программирования на языке С+ [Электронный ресурс] : учебное пособие / В.Н. Васильев. — Электрон. текстовые данные. — Волгоград: Волгоградский институт бизнеса, Вузовское образование, 2010. — 72 с. Режим доступа: http://www.iprbookshop.ru/11341.html
2.	Никишев, В. К. Современные языки программирования : лабораторный практикум / В. К. Никишев ; [отв. ред. В. П. Желтов] ; Чуваш. гос. ун-т им. И. Н. Ульянова. - Чебоксары : Изд-во Чуваш. ун-та, 2015. - 167с.
3.	Ильин, Д. В. Язык программирования С++ : курс лекций [для 1 курса факультета информатики и вычислительной техники] / Д. В. Ильин, Л. А. Ильина ; [отв. ред. И. Т. Артемьев] ; Чуваш. гос. ун-т им. И. Н. Ульянова. - Чебоксары : Изд-во Чуваш. ун-та, 2015. - 124с.

7.2. Рекомендуемая дополнительная литература

№	Название
1.	Ильин, Д. В. Языки программирования : лабораторный практикум [для 1 курса факультета информатики и вычислительной техники] / Д. В. Ильин, Л. А. Ильина ; [отв. ред. И. Т. Артемьев] ; Чуваш. гос. ун-т им. И. Н. Ульянова. - Чебоксары : Изд-во Чуваш. ун-та, 2015. - 55с.
2.	Ковалевская Е.В. Методы программирования [Электронный ресурс] : учебное пособие / Е.В. Ковалевская, Н.В. Комлева. — Электрон. текстовые данные. — М. : Евразийский открытый институт, 2011. — 320 с. Режим доступа: http://www.iprbookshop.ru/10784.html

7.3. Программное обеспечение, профессиональные базы данных, информационно-справочные системы.

Программное обеспечение, профессиональные базы данных, информационно-справочные системы, предоставляемые управлением информатизации ФГБОУ ВО «Чувашский государственный университет имени И.Н. Ульянова» доступны по ссылке <http://ui.chuvsu.ru/>*

7.3.1. Программное обеспечение

№ п/п	Наименование	Условия доступа/скачивания
1.	MS Office/ LibreOffice	лицензия университета/ свободное лицензионное соглашение (https://ru.libreoffice.org/)
2.	MS Windows/Linux (Ubuntu)	лицензия университета/ свободное лицензионное соглашение (http://ubuntu.ru/)
3.	Visual Studio Community	http://www.visualstudio.com/ru/vs/community

7.3.2. Базы данных, информационно-справочные системы

№ п/п	Наименование программного обеспечения	Условия доступа/скачивания
1.	Гарант	из внутренней сети университета (договор)*
2.	Консультант +	

7.3.3. Рекомендуемые интернет-ресурсы и открытые он-лайн курсы

№ п/п	Наименование интернет ресурса	Режим доступа
1.	Российская Государственная Библиотека	http://www.rsl.ru
2.	Государственная публичная научно-техническая библиотека России	http://www.gpntb.ru
3.	Фундаментальная библиотека Нижегородского государственного университета	http://www.unn.ru/library
4.	Научная библиотека Казанского государственного университета	http://lsl.ksu.ru
5.	Научная электронная библиотека	http://elibrary.ru
6.	Полнотекстовая библиотека учебных и учебно-методических материалов	http://window.edu.ru
7.	Электронно-библиотечная система IPRbooks	http://www.iprbookshop.ru

8. Материально-техническое обеспечение дисциплины

Учебные аудитории для лекционных занятий по дисциплине оснащены автоматизированным рабочим местом (АРМ) преподавателя, обеспечивающим тематические иллюстрации и демонстрации, соответствующие программе дисциплины в составе:

- ПЭВМ с доступом в интернет (операционная система, офисные программы, антивирусные программы);
- мультимедийный проектор с дистанционным управлением;
- настенный экран.

Учебные аудитории для самостоятельных занятий по дисциплине оснащены АРМ преподавателя и пользовательскими АРМ по числу обучающихся, объединенных локальной сетью («компьютерный» класс), с возможностью подключения к сети Интернет и доступом к электронной информационно-образовательной среде ФГБОУ ВО «Чувашский государственный университет имени И.Н. Ульянова».

9. Средства адаптации преподавания дисциплины к потребностям лиц с ограниченными возможностями

В случае необходимости, инвалидам и лицам с ограниченными возможностями здоровья могут в случае необходимости, обучающимся из числа лиц с ограниченными возможностями здоровья (по заявлению обучающегося) могут предлагаться одни из следующих вариантов восприятия информации с учетом их индивидуальных психофизических особенностей, в том числе с применением электронного обучения и дистанционных технологий:

- для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; в форме аудиофайла (перевод учебных материалов в аудиоформат); в печатной форме на языке Брайля; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания и консультации.

- для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; видеоматериалы с субтитрами; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания и консультации.

- для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; в форме аудиофайла; индивидуальные задания и консультации.

Кроме того, могут применяться элементы дистанционных образовательных технологий для изучения учебного материала на удалении.

10. Методические рекомендации по освоению дисциплины

В ходе лекционных занятий студенту рекомендуется вести конспектирование учебного материала. Следует обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации, положительный опыт в ораторском искусстве. При составлении конспекта желательно оставлять в рабочих конспектах поля, на которых в дальнейшем можно делать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений. В ходе лекционных занятий рекомендуется задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций.

В ходе подготовки к лабораторным работам рекомендуется изучить основную литературу, ознакомиться с дополнительной литературой, новыми публикациями в периодических изданиях: журналах, газетах и т.д. основой для выполнения лабораторной работы являются разработанные кафедрой методические указания. Рекомендуется дорабатывать конспект лекций, делая в нем соответствующие записи из литературы, рекомендованной преподавателем и предусмотренной рабочей программой дисциплины. Желательно подготовить тезисы для выступлений по всем учебным вопросам, выносимым на лабораторное занятие. Готовясь к докладу или реферативному сообщению, рекомендуется обращаться за методической помощью к преподавателю, составить план-конспект своего выступления, продумать примеры с целью обеспечения тесной связи изучаемой теории с практикой. В процессе подготовки студент может дополнить список использованной литературы современными источниками, не представленными в списке рекомендованной литературы, и в дальнейшем использовать собственные подготовленные учебные материалы при написании курсового проекта.

Формы организации студентов на лабораторных работах индивидуальная. При индивидуальной форме организации занятий каждый студент выполняет индивидуальное задание.

Качество выполнения лабораторных работ является важной составляющей оценки текущей успеваемости обучающегося.

№ п/п	Прилагаемый к Рабочей программе дисциплины документ, содержащий текст обновления	Решение кафедры вычислительной техники		Подпись заведующего кафедрой	И. О.Фамилия заведующего кафедрой
		Дата	протокол №		
1	Приложение № 1				
2	Приложение № 2				
3	Приложение № 3				
4	Приложение № 4				
5	Приложение № 5				