

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 компьютерных технологий

«УТВЕРЖДАЮ»
Проректор по учебной работе

И.Е. Поверинов

«31» августа 2017 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
«ВВЕДЕНИЕ В ПРОФЕССИОНАЛЬНУЮ ДЕЯТЕЛЬНОСТЬ ПРОГРАММИСТА»

Направление подготовки (специальность) 09.03.03 Прикладная информатика
Квалификация (степень) выпускника – бакалавр
Профиль (направленность) Прикладная информатика в дизайне
Прикладной бакалавриат

Чебоксары – 2017

Рабочая программа основана на требованиях Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.03 «Прикладная информатика», утвержденного приказом Министерства образования и науки Российской Федерации № 207 от 12.03.2015 г.

СОСТАВИТЕЛЬ (СОСТАВИТЕЛИ):

доцент, канд. техн. наук, доцент

 А.Н. Ванюлин

старший преподаватель

 Е.А. Мытникова

ОБСУЖДЕНО:

на заседании кафедры компьютерных технологий «30» августа 2017г., протокол № 1

заведующий кафедрой

 Т.А. Лавина

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.....	5
4.3. Объем дисциплины, виды учебной работы обучающихся по заочной форме обучения .	6
5. Содержание разделов дисциплины	6
5.1. Лекции	6
5.2. Лабораторные работы	7
5.3. Вопросы для самостоятельной работы студента в соответствии с содержанием разделов дисциплины	7
6. Образовательные технологии	7
7. Формы аттестации и оценочные материалы	8
7.1. Вопросы к зачету	8
7.2. Вопросы и задачи к экзамену	10
7.3. Выполнение и примерная тематика курсового проекта	10
7.4. Выполнение и примерные задания расчетно-графической работы.....	10
7.5. Выполнение и примерная тематика контрольной работы.....	10
8. Учебно-методическое и информационное обеспечение дисциплины.....	12
8.1. Рекомендуемая основная литература	12
8.2. Рекомендуемая дополнительная литература (печатная, в том числе методические указания).....	12
8.3. Рекомендуемые методические разработки по дисциплине.....	12
8.4. Программное обеспечение.....	12
8.5. Профессиональные базы данных, информационно-справочные системы	13
8.6. Рекомендуемые интернет-ресурсы и открытые он-лайн курсы.....	13
9. Материально-техническое обеспечение дисциплины.....	13
10. Средства адаптации преподавания дисциплины к потребностям лиц с ограниченными возможностями.....	13
11. Методические рекомендации по освоению дисциплины	13
12. Методические указания для преподавателя по проведению лабораторных работ	14

1. Цель и задачи обучения по дисциплине

Целью изучения дисциплины является обучение студентов методике разработки программных средств с использованием технологий объектно-ориентированного программирования и обобщенного программирования.

Задачи изучения дисциплины:

- моделирование прикладных и информационных процессов, описание реализации информационного обеспечения прикладных задач;
- программирование в ходе разработки информационной системы;
- проведение работ по инсталляции программного обеспечения информационных систем и загрузке баз данных.

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

Дисциплина является дисциплиной вариативной части Блока 1. Дисциплины (модули).

Дисциплины и практики учебного плана, изученные (изучаемые) обучающимися и формирующие входные знания и умения для обучения по данной дисциплине: Информатика и программирование, Программирование на языках высокого уровня, Программная инженерия.

Дисциплины и практики учебного плана, которые предстоит изучить (изучаются) обучающимся и для которых при обучении по данной дисциплине формируются входные знания и умения: Проектный практикум, Современные языки программирования, Производственная практика (практика по получению профессиональных умений и опыта профессиональной деятельности), Преддипломная практика для 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:

обще профессиональных (ОПК):

- способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности (ОПК-4);

профессиональных (ПК):

- способностью программировать приложения и создавать программные прототипы решения прикладных задач (ПК-8);
- способностью составлять техническую документацию проектов автоматизации и информатизации прикладных процессов (ПК-9);
- способностью осуществлять инсталляцию и настройку параметров программного обеспечения информационных систем (ПК-13).

В результате обучения по дисциплине обучающийся должен (ЗУН):

знать:

- 31 – концепции объектно-ориентированного программирования (ПК-8);
- 32 – основные понятия объектно-ориентированного программирования (ПК-8);
- 33 – шаблоны STL (ПК-8);
- 34 – алгоритмы решения стандартных задач в области обработки информации на основе информационной культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности (ОПК-4).
- 35 – техническую документацию (ПК-9);
- 36 – параметры программного обеспечения (ПК-13);

уметь:

- У1 – описывать классы (ПК-8);

- У2 – использовать шаблоны библиотеки STL(ПК-8);
- У3 – анализировать варианты использования алгоритмов решения стандартных задач в области обработки информации и с учетом требований информационной безопасности (ОПК-4);
- У4 – составлять техническую документацию проектов автоматизации и информатизации прикладных процессов (ПК-9);
- У5 – осуществлять инсталляцию и настройку параметров программного обеспечения информационных систем (ПК-13).

владеть навыками:

- Н1 – объектно-ориентированного программирования (ПК-8);
- Н2 – обобщенного программирования (ПК-8);
- Н3 – навыками и приемами использования алгоритмов решения стандартных задач в области обработки информации на основе информационной культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности (ОПК-4).
- Н4 – составления технической документации (ПК-9)
- Н5 – осуществления инсталляции и настройки параметров программного обеспечения информационных систем (ПК-13).

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 (лабораторные работы), групповые и (или) индивидуальные консультации, в том числе в электронной информационно-образовательной среде.

Обозначения:

Л – лекции, л/р – лабораторные работы, п/р – практические занятия, 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. Программирование на C++	ОПК-4, ПК-8, ПК-13, ПК-9	31, 32, 34, 35, 36 У1, У4, У5 Н1, Н4, Н5
1.1. Раздельная компиляция и пространства имен		
1.2. Классы. Объекты		
1.3. Классы. Перегрузка операций.		
1.4. Иерархия классов, наследование.		
Раздел 2. Шаблоны STL	ОПК-4, ПК-8	33, У2, У3, Н2, Н3
2.1. Итераторы, алгоритмы, объекты функции (функторы)		
2.2. Контейнеры последовательности		
2.3. Ассоциативные контейнеры		
2.4. Контейнеры-адаптеры		
Зачет	ОПК-4, ПК-8, ПК-13, ПК-9	31 - 36, У1 – У5, Н1 – Н5

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

Содержание	Всего, час	Контактная работа, в том числе в электронной информационно-образовательной среде час.	СРС, час	ИФР, час	К, час

		Л	л/р	п/р	КСР			
Раздел 1. Программирование на C++	18	8				10	8	
1.1. Раздельная компиляция и пространства имен	3	1				2	1	
1.2. Классы. Объекты	4	2				2	2	
1.3. Классы. Перегрузка операций.	4	2				2	2	
1.4. Иерархия классов, наследование.	7	3				4	3	
Раздел 2. Шаблоны STL	52	8	32			12	8	
2.1. Итераторы, алгоритмы, объекты функции (функторы)	14	2	8			4	2	
2.2. Контейнеры последовательности	14	2	8			4	2	
2.3. Ассоциативные контейнеры	12	2	8			2	2	
2.4. Контейнеры-адаптеры	12	2	8			2	2	
Зачет	2				2			
Итого	72	16	32		2	22	16	
Зачетных единиц	2							

Вид промежуточной аттестации: зачет в 5 семестре.

4.3. Объем дисциплины, виды учебной работы обучающихся по заочной форме обучения

Содержание	Всего, час	Контактная работа, в том числе в электронной информационно-образовательной среде час				СРС, час	ИФР, час	К, час
		Л	л/р	п/р	КСР			
Раздел 1. Программирование на C++	18	2				16		
1.1. Раздельная компиляция и пространства имен	2					2		
1.2. Классы. Объекты	5	1				4	1	
1.3. Классы. Перегрузка операций.	5					5		
1.4. Иерархия классов, наследование.	6	1				5	1	
Раздел 2. Шаблоны STL	45	4	6			35		
2.1. Итераторы, алгоритмы, объекты функции (функторы)	9	1				8	1	
2.2. Контейнеры последовательности	12	1	2			9	1	
2.3. Ассоциативные контейнеры	12	1	2			9	1	
2.4. Контейнеры-адаптеры	12	1	2			9	1	
Контрольная работа	6					6		
Зачет	3							3
Итого	72	6	6			57	6	3
Зачетных единиц	2							

5. Содержание разделов дисциплины

5.1. Лекции

Раздел 1. Программирование на C++

Тема 1.1. Раздельная компиляция и пространства имен

Пространства имен как средство реализации модульности. Поиск имен и разрешение конфликтов. Объединение пространств имен. Принципы дизайна пакетов.

Тема 1.2. Классы. Объекты

Составные части объектного подхода. Абстрагирование. Инкапсуляция. Модульность. Иерархия. Типизация. Параллелизм. Сохраняемость. Объектно-ориентированная модель. Понятие объекта. Свойства, присущие объектам. Состояние. Поведение. Идентичность. Структура объявления класса. Доступ к членам класса. Поля данных класса как механизм реализации состояния объекта. Функции члены класса как механизм реализации поведения объекта. Спецификаторы доступа для обеспечения инкапсуляции. Средства управления жизнью объ-

екта. Конструкторы и деструкторы. Конструирование и уничтожение объектов и массивов объектов.

Тема 1.3. Классы. Перегрузка операций.

Перегрузка операторов C++ как реализация поведения с предопределенным смыслом. Дружественность как механизм нарушения инкапсуляции. Достоинства и недостатки механизма дружественности.

Тема 1.4. Иерархия классов, наследование.

Одиночное наследование. Понятие производного класса. Управление доступом в производных классах. Конструкторы и деструкторы, совмещение имен методов при наследовании, иерархии. Абстрактные классы и виртуальные функции. Виртуальный полиморфизм.

Проблема множественного наследования. Виртуальное наследование как средство разрешения коллизий. Порядок вызовов конструкторов и деструкторов при множественном наследовании. Чистые виртуальные классы, понятие интерфейса.

Раздел 2. Шаблоны STL

Тема 2.1. Итераторы, алгоритмы, объекты функции (функторы)

Общие сведения (понятия контейнеров, итераторов и объектов- функций) Контейнеры (виды контейнеров, последовательные и ассоциативные контейнеры, адаптеры). Итераторы (итератор как обобщение указателя, классы итераторов). Алгоритмы (примеры алгоритмов с использованием итераторов: алгоритмы сортировки, алгоритмы, не изменяющие содержание контейнера, алгоритмы, изменяющие содержание контейнера).

Тема 2.2. Контейнеры последовательности

vector, array, deque, list, forward_list

Тема 2.3. Ассоциативные контейнеры

map, unordered_map, set, unordered_set, multimap, multiset, unordered_multimap, unordered_multiset

Тема 2.4. Контейнеры-адаптеры

queue, priority_queue, stack

5.2. Лабораторные работы

Тема	Трудоемкость, час
Лабораторная работа №1. <i>Итераторы, алгоритмы, объекты функции (функторы)</i>	8
Лабораторная работа № 2. <i>Контейнеры последовательности</i>	8
Лабораторная работа № 3. <i>Ассоциативные контейнеры</i>	8
Лабораторная работа № 4. <i>Контейнеры-адаптеры</i>	8

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

1. Принципы дизайна пакетов.
2. Средства управления жизнью объекта.
3. Дружественность как механизм нарушения инкапсуляции.
4. Виртуальное наследование как средство разрешения коллизий.

6. 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.);

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

Интерактивные технологии:

№ темы	Вид занятия	Используемые интерактивные технологии
1.1–2.4	Лекция	Презентации, разбор конкретных ситуаций

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 (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: систематическая проработка конспектов лекций и учебной литературы; подготовка к выполнению лабораторных работ, оформление отчетов и подготовка к защите выполненных работ; выполнение контрольной работы (для студентов заочной формы обучения).

Формами контроля самостоятельной работы выступают: оценивание устного ответа студента на занятии, оценивание письменных отчетов по результатам выполненных заданий и лабораторных работ, проверка контрольной работы (для студентов заочной формы обучения). Результаты самостоятельной работы учитываются при оценке знаний на зачете.

7. 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 зачета. Принимается зачет преподавателем, читающим лекции по данной учебной дисциплине в соответствии с перечнем основных вопросов, выносимых для контроля знаний обучающихся:

7.1. Вопросы к зачету

1. Пространства имен как средство реализации модульности.
2. Поиск имен и разрешение конфликтов.
3. Объединение пространств имен.
4. Принципы дизайна пакетов.
5. Составные части объектного подхода.

6. Абстрагирование.
7. Инкапсуляция.
8. Модульность.
9. Иерархия.
10. Типизация.
11. Параллелизм.
12. Сохраняемость.
13. Объектно-ориентированная модель.
14. Понятие объекта.
15. Свойства, присущие объектам.
16. Состояние.
17. Поведение.
18. Идентичность.
19. Структура объявления класса.
20. Доступ к членам класса.
21. Поля данных класса как механизм реализации состояния объекта.
22. Функции члены класса как механизм реализации поведения объекта.
23. Спецификаторы доступа для обеспечения инкапсуляции.
24. Средства управления жизнью объекта.
25. Конструкторы и деструкторы.
26. Конструирование и уничтожение объектов и массивов объектов.
27. Перегрузка операторов C++ как реализация поведения с предопределенным смыслом.
28. Дружественность как механизм нарушения инкапсуляции.
29. Достоинства и недостатки механизма дружественности.
30. Одиночное наследование.
31. Понятие производного класса.
32. Управление доступом в производных классах.
33. Конструкторы и деструкторы, совмещение имен методов при наследовании, иерархии.
34. Абстрактные классы и виртуальные функции.
35. Виртуальный полиморфизм.
36. Проблема множественного наследования.
37. Виртуальное наследование как средство разрешения коллизий.
38. Порядок вызовов конструкторов и деструкторов при множественном наследовании.
39. Чистые виртуальные классы, понятие интерфейса.
40. Понятия контейнеров, итераторов и объектов- функций
41. Контейнеры (виды контейнеров, последовательные и ассоциативные контейнеры, адаптеры).
42. Итераторы (итератор как обобщение указателя, классы итераторов).
43. Алгоритмы (примеры алгоритмов с использованием итераторов: алгоритмы сортировки, алгоритмы, не изменяющие содержание контейнера, алгоритмы, изменяющие содержание контейнера).
44. vector
45. array
46. deque
47. list
48. forward_list
49. map
50. unordered_map
51. set
52. unordered_set

53. multimap multiset
54. unordered_multimap
55. unordered_multiset
56. queue
57. priority_queue
58. stack

Зачет проводится по окончании занятий по дисциплине до начала экзаменационной сессии.

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

Критерии получения зачета по дисциплине:

– Оценка «зачтено» проставляется студенту, выполнившему и защитившему в полном объеме лабораторные работы в течение семестра, выполнившему контрольную работу (у заочной формы обучения), чей уровень знаний, умений и навыков соответствует уровню оценок «отлично», «хорошо» или «удовлетворительно».

– Оценка «не зачтено» проставляется студенту, не выполнившему и (или) не защитившему в полном объеме лабораторные работы в течение семестра, не выполнившему контрольную работу (у заочной формы обучения), либо чей уровень знаний, умений и навыков соответствует уровню оценки «неудовлетворительно».

Общими критериями, определяющими оценку знаний, умений и навыков 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.

7.2. Вопросы и задачи к экзамену

Не предусмотрены.

7.3. Выполнение и примерная тематика курсового проекта

Не предусмотрены.

7.4. Выполнение и примерные задания расчетно-графической работы

Не предусмотрены.

7.5. Выполнение и примерная тематика контрольной работы

Контрольная работа выполняется студентами заочной формы обучения в процессе изучения дисциплины. Общее руководство и контроль за ходом выполнения контрольной работы осуществляет преподаватель соответствующей дисциплины. Контрольная работа выполняется в соответствии с методическими указаниями для обучающихся.

Основными функциями руководителя контрольной работы являются:

– определение и формулирование задания контрольной работы;

- консультирование по вопросам содержания и последовательности выполнения контрольной работы;
- оказание помощи студенту в подборе необходимой литературы;
- контроль хода выполнения контрольной работы.

Примерная тематика контрольных работ:

Задача 1. Создание простого класса:

Создать класс «транспортное средство». Информация о транспортном средстве – это: название, мощность двигателя, максимальная скорость. Функции:

- а) ввод данных о транспортном средстве;
- б) вывод на экран информации о транспортном средстве;
- в) функция, возвращающая название транспортного средства;
- г) функция, возвращающая информацию о мощности двигателя;
- д) функция, возвращающая максимальную скорость транспортного средства.

Создать массив объектов класса «транспортное средство». Подсчитать среднюю скорость транспортных средств. Вывести названия транспортных средств с мощностью двигателя выше заданной.

Задача 2. Перегрузка операторов:

Матрица – KMatrix

Методы:

конструкторы, деструктор;

операции: [], =, ==, !=, +, +=, -, -=, *, *=, <<, >>

функции:

Transpose – транспонирование матрицы,

Rows() – количество строк,

Cols() – количество колонок,

Swap() - обменять значения

Задача 3. Абстрактные классы. Виртуальные функции:

Класс: место - область - город - мегаполис

1. Определить иерархию системы, которая заключается в построении базового абстрактного класса и нескольких обоснованных производных или связанных классов.

2. Определить в базовом абстрактном классе чисто виртуальную функцию.

3. Каждый класс входящий в иерархию, использует открытое подражания, должен содержать по крайней мере одну виртуальную функцию. Функциональность указанных функций не должна дублироваться.

4. Предусмотреть 2 возможных исключительные ситуации работы с динамической памятью или с работой файлами (с демонстрацией работы через try-catch).

Задача 4. Задачи с использованием STL:

Реализуйте структуру данных типа “множество строк”. Хранимые строки – непустые последовательности длиной не более 10 символов, состоящие из строчных латинских букв. Структура данных должна поддерживать операции добавления строки в множество, удаления строки из множества и проверки принадлежности данной строки множеству. Максимальное количество элементов в хранимом множестве не превосходит 10^6 .

Оценивание контрольной работы осуществляется в соответствии с полнотой и качеством выполнения задания на работу, качеством защиты работы (ответы на вопросы, презентация и др.). Оценка работы отражает уровень сформированности соответствующих компетенций:

- «зачтено» – работа выполнена в соответствии с утвержденным планом и заданием,

полностью раскрыто содержание каждого вопроса; решены все задачи; сделаны верные выводы; оформление работы соответствует предъявляемым требованиям; при защите работы обучающийся демонстрирует владение материалом;

– «не зачтено» – если работа не удовлетворяет хотя бы одному из требований, предыдущего абзаца.

8. 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/>

8.1. Рекомендуемая основная литература

№ п/п	Наименование
1.	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
2.	Стенли Липпман Язык программирования C++ [Электронный ресурс] : полное руководство / Липпман Стенли, Лажойе Жози. — Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 1104 с. - Режим доступа: http://www.iprbookshop.ru/63964.html
3.	Васильев А.Н. Объектно-ориентированное программирование на C++ [Электронный ресурс] / А.Н. Васильев. — Электрон. текстовые данные. — СПб. : Наука и Техника, 2016. — 544 с. - Режим доступа: http://www.iprbookshop.ru/60648.html

8.2. Рекомендуемая дополнительная литература (печатная, в том числе методические указания)

№ п/п	Наименование
1.	Никишев, В. К Объектно-ориентированное программирование : учебное пособие / В. К. Никишев ; [отв. ред. В. П. Желтов] ; Чуваш. гос. ун-т им. И. Н. Ульянова. - Чебоксары : Изд-во Чуваш. ун-та, 2016. - 262с.
2.	Борисенко В.В. Основы программирования [Электронный ресурс] / В.В. Борисенко. – Электрон. текстовые данные. – М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. – 323 с. – 978-5-9556-00039-0. – Режим доступа: http://www.iprbookshop.ru/52206.html
3.	Обломов, И. А. Объектно-ориентированное программирование : лабораторный практикум / И. А. Обломов ; [отв. ред. А. Л. Симаков] ; Чуваш. гос. ун-т им. И. Н. Ульянова. - Чебоксары : Изд-во Чуваш. ун-та, 2014. - 111с.

8.3. Рекомендуемые методические разработки по дисциплине

№ п/п	Наименование	Условия доступа
	Введение в профессиональную деятельность программиста: лабораторный практикум	URL: http://moodle.chuvsu.ru/course/view.php?id=1220

8.4. Программное обеспечение

Доступное программное лицензионное и свободно распространяемое программное обеспечение, предоставляемое студенту университетом возможно для загрузки и использования по URL: http://ui.chuvsu.ru/index.php/2010-06-25-10-45-35*.

№ п/п	Наименование Рекомендуемого ПО	Условия доступа/скачивания
		свободное лицензионное соглашение:
1.	Microsoft Visual Studio	https://www.microsoft.com/ru-ru/SoftMicrosoft/vs2017
2.	DevC++	https://sourceforge.net/projects/orwelldevcpp/
3.	Linux/ Ubuntu	http://ubuntu.ru/
4.	LibreOffice	https://ru.libreoffice.org/
		из внутренней сети университета (договор)*
1.	Microsoft Windows	
2.	Microsoft Office	

8.5. Профессиональные базы данных, информационно-справочные системы

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

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

№ п/п	Наименование интернет ресурса	Режим доступа
	Открытое образование. Языки программирования. Генезис и практики	URL: https://openedu.ru/course/ITMOUniversity/PROLAN/
2.	Национальный открытый университет. Язык программирования C++	URL: http://www.intuit.ru/studies/courses/17/17/info
3.	Национальный открытый университет. Основы объектно-ориентированного программирования	URL: http://www.intuit.ru/studies/courses/71/71/info

9. 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 лабораторных и 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».

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

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.

11. 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. Основой для выполнения лабораторной работы являются разработанные кафедрой методические указания. Рекомендуется дорабатывать свой конспект лекций, делая в нем соответствующие записи из литературы, рекомендованной преподавателем и предусмотренной рабочей программой дисциплины. 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 подготовке к зачету, выполнении контрольной работы.

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

Если в результате выполнения лабораторной работы запланирована подготовка письменного отчета, то отчет о выполненной работе необходимо оформлять в соответствии с требованиями методических указаний. Качество выполнения лабораторных работ является важной составляющей оценки текущей успеваемости обучающегося.

12. Методические указания для преподавателя по проведению лабораторных работ

Цель лабораторных работ

Лабораторные работы по курсу «Введение в профессиональную деятельность программиста» направлены на обучение студентов методике разработки программных средств с использованием технологий объектно-ориентированного программирования и обобщенного программирования.

Инструментальной платформой курса являются Microsoft Visual Studio C++ и Dev-C++.

Технология проведения занятий

Для лабораторных работ студент выбирает свой вариант. Во всех работах он должен использовать методические указания к лабораторным работам, которые находятся у преподавателя в электронном виде, а также в Системе дистанционного обучения ЧГУ им. И. Н. Ульянова.

Методика проведения занятий

Лабораторные работы выполняются в следующей последовательности

1. Прочитать теоретическую часть.
 2. Рассмотреть разобранный пример.
 3. Определить свой номер варианта.
 4. Выполнить работу.
 5. Показать преподавателю, как выполнялась работа.
 6. Составить отчет в электронном виде.
 7. Ответить на контрольные вопросы.
- По каждой лабораторной работе студент оформляет итоговый отчет.

Содержание отчета:

1. Титульный лист.
2. Теоретические сведения по темам лабораторных работ.
3. Задания по варианту.
4. Последовательность выполнения каждой работы.
5. Результаты выполнения каждой работы.
6. Выводы.
7. Использованная литература.

Методы обучения и способы учебной деятельности

Для подготовки могут быть использованы находящиеся у преподавателя в электронном виде литературные источники. При выполнении лабораторных работ для понимания студент может пользоваться встроенной справкой применяемых программ, поисковыми системами и другими ресурсами Интернета.

Студенты могут консультироваться по вопросам выполнения лабораторных работ как у преподавателя, так и у других студентов.