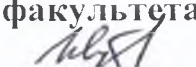
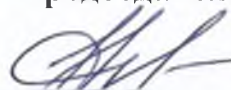


Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Дагестанский государственный технический университет»

РЕКОМЕНДОВАНО
К УТВЕРЖДЕНИЮ:
Декан, председатель совета
факультета КТВТиЭ,
 Юсуфов Ш.А.

«13» 09 2018г.

УТВЕРЖДАЮ:
Проректор по учебной работе,
председатель методического
совета ДГТУ
 Суракатов Н.С.

«13» 09 2018г.

РАБОЧАЯ ПРОГРАММА

Дисциплина Б1.В.ОД.11 Объектно-ориентированное программирование
для направления 09.03.04 ПРОГРАММНАЯ ИНЖЕНЕРИЯ
по профилю Разработка программно-информационных систем
факультет Компьютерных технологий, вычислительной техники и энергетики
наименование факультета, где ведется дисциплина
кафедра Программное обеспечение вычислительной техники и
автоматизированных систем
наименование кафедры, за которой закреплена дисциплина

Квалификация выпускника (степень) бакалавр
Форма обучения очная; курс 4; семестр(ы) 7;
Всего трудоемкость в зачетных единицах (часах) 5 зет(180);
Лекции 34 (час); Экзамен 7 (сем); 13ЕТ-36ч;
Практические (семинарские) занятия - (час); Зачет - (семестр);
Лабораторные занятия 34 (час);
Курсовой проект (работа, РГР) 7 (семестр);
Самостоятельная работа 76 (час).

Зав. кафедрой


подпись

/Мелехин В.Б./
ФИО

Начальник УО



подпись

/Магомаева Э.В./
ФИО

Программа составлена в соответствии с требованиями ФГОС ВО с учетом рекомендаций ООП ВО по направлению 09.03.04 «Программная инженерия» и профилю подготовки «Разработка программно-информационных систем».

Программа одобрена на заседании выпускающей кафедры от 12.09.2018 года, протокол № 1.

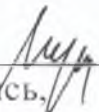
Зав. выпускающей кафедрой по данному направлению (специальности,

профилю)  /Мелехин В.Б./

ОДОБРЕНО:

Методической комиссией по УГС
направления подготовки
09.00.00 «Информатика и вычислительная
техника»
шифр и полное наименование
09.03.04 «Программная инженерия»
направления

Председатель МК

 /Абдулгалимов А.М./
подпись. ФИО

« 12 » 09 2018г.

АВТОР ПРОГРАММЫ:

Камилова А.М.,
ФИО, уч. степень, ученое
звание, подпись
ст. преподаватель



1. Цели освоения дисциплины.

Учебная дисциплина «Объектно-ориентированное программирование» реализует требования федерального государственного образовательного стандарта высшего образования по направлению бакалавриата 09.03.04 «Программная инженерия» по профилю «Разработка программно-информационных систем».

Цель дисциплины – ознакомить обучаемых с основными тенденциями и направлениями развития современных технологий программирования и обработки данных, с основными принципами и методологией разработки объектно-ориентированного программного обеспечения.

Задача дисциплины - привить обучаемым навыки использования методов объектно-ориентированного программирования при разработке программных систем разного уровня сложности.

2. Место дисциплины в структуре ООП бакалавриата.

«Объектно-ориентированное программирование» входит в цикл обязательных дисциплин (вариативная часть).

Для успешного усвоения данной дисциплины необходимо, чтобы обучаемый владел знаниями, умениями и навыками, сформированными в процессе изучения дисциплин “Теоретическая информатика”, “Информатика и программирование” и “Программирование”.

Знания, полученные обучаемыми по дисциплине «Объектно-ориентированное программирование», могут быть использованы при изучении дисциплин базового цикла «Проектирование и архитектура программных систем», “Построение человеко-машинного интерфейса” и «Тестирование программного обеспечения».

3. Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) «Объектно-ориентированное программирование».

Процесс изучения дисциплины направлен на формирование у обучаемого следующих компетенций:

- способностью к самоорганизации и самообразованию (ОК-7);
- готовностью применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов (ОПК-3);
- готовностью применять основные методы и инструменты разработки программного обеспечения (ПК-1);
- владением навыками использования различных технологий разработки программного обеспечения (ПК-3);

В результате изучения дисциплины обучаемый должен:

знать:

- основные принципы объектно-ориентированного программирования;
- принципы построения классов и критерии проверки правильности построения классов;

- основные тенденции и направления в области развития технологий объектно-ориентированного программирования.

уметь:

- ставить задачу и разрабатывать алгоритм ее решения, использовать прикладные программные системы программирования, разрабатывать основные программные документы;

- работать с современными объектно-ориентированными системами программирования.

владеть:

- языками объектно – ориентированного программирования, навыками разработки и отладки программ не менее, чем на одном из объектно – ориентированных языков программирования.

4. Структура и содержание дисциплины «Объектно-ориентированное программирование»

4.1. Содержание дисциплины

№ п/п	Раздел дисциплины Тема лекции и вопросы	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Формы текущего контроля (по срокам текущей аттестации)
				ЛК	ПЗ	ЛР	СРС	
1	2	3	4	5	6	7	8	9
1.	<p>Лекция 1. Тема: «Составные типы данных в С++. Динамическое распределение памяти». Объявление функций в С++. Способы передачи параметров функциям. Составные типы данных в С++: массивы, структуры и объединения. Указатели в С++. Работа с массивами и структурами с помощью указателей. Динамическое распределение памяти.</p>	7	1	2		4	2	Входной контроль
2.	<p>Лекция 2. Тема: «ООП как направление развития теории и практики создания ПО». Основные этапы развития технологий программирования. Причины появления и актуальность ООП. Преимущества и недостатки ООП. Области применения ООП. Основные принципы ООП. Объектно-ориентированные языки программирования.</p>	7	2	2			4	КР№1
3.	<p>Лекция 3. Тема: «Понятие класса и объекта в С++». Назначение и семантика классов. Классы и абстрактные типы данных. Защита данных на уровне класса (модификаторы доступа). Объекты ООП и объекты реального мира.</p>	7	3	2			4	

	Представление данных. Классификация методов. Инкапсуляция данных.							
4.	Лекция 4. Тема: «Конструкторы и деструкторы в С++». Создание объектов и массивов объектов. Правила для конструкторов и деструкторов в С++. Стандартный конструктор, конструктор с параметрами. Список инициализации. Конструктор копирования. Указатель this.	7	4	2		4	8	
5.	Лекция 5. Тема: «Конструкторы и деструкторы в С++». Конструктор копирования. Глубинное и поверхностное копирование. Указатель this. Встраиваемые функции (inline).	7	5	2				
6.	Лекция 6. Тема: «Члены класса, общие для всех экземпляров. Константные данные класса в С++». Назначение членов класса, общих для всех объектов класса. Статические переменные и статические функции класса (static). Константные переменные и функции класса (const).	7	6	2			6	
7.	Лекция 7. Тема: «Друзья класса (friend) в С++». Функции, дружественные одному классу, а также нескольким классам. Методы класса, дружественные другому классу. Дружественные классы.	7	7	2		4	8	КР№2
8.	Лекция 8. Тема: «Механизмы наследования в С++». Простое наследование. Спецификаторы доступа при наследовании. Конструкторы и деструкторы производных классов (очередность вызова). Переопределение и вызов элементов	7	8	2			8	

	базового класса. Множественное наследование. Виртуальные базовые классы.							
9.	Лекция 9. Тема: «Механизмы наследования в С++». Множественное наследование. Виртуальные базовые классы.	7	9	2				
10	Лекция 10. Тема: «Перегрузка в С++». Назначение перегрузки. Перегрузка функций. Декодирование компилятором имен функций. Перегрузка конструкторов. Неоднозначность, возникающая при перегрузке.	7	10	2		4	8	
11	Лекция 11. Тема: «Перегрузка в С++». Перегрузка операторов. Правила перегрузки операторов. Перегрузка унарных и бинарных операторов.	7	11	2				КР№3
12	Лекция 12. Тема: «Полиморфизм и виртуальные функции в С++». Виды и реализации полиморфизма. Механизмы раннего и позднего связывания. Динамический полиморфизм. Виртуальные функции. Виртуальные деструкторы. Чисто виртуальные функции и абстрактные классы.	7	12	2		4	8	
13	Лекция 13. Тема: «Параметризованные типы в С++». Шаблоны функций. Специализация шаблонов функций. Шаблоны классов. Специализация шаблонов класса. Стандартная библиотека шаблонов STL. Перегрузка шаблонов. Контейнеры и алгоритмы библиотеки STL	7	13	2		4	4	
14	Лекция 14. Тема: «Стандартная библиотека шаблонов С++». Назначение и состав библиотеки. Контейнеры: последовательные и ассоциативные. Универсальные	7	14	2			4	

	алгоритмы.								
15	Лекция 15. Тема: «Классы VCL-библиотеки». Иерархия классов библиотеки. Объекты, компоненты, элементы управления. Классы командных кнопок, флажков и радиокнопок. Классы текстовых компонентов, списков и панелей. Классы стандартных диалогов VCL-библиотеки и обработка событий.	7	15	2		4	4		
16	Лекция 16. Тема: «Объектно-ориентированное программирование на C# и Java. Сходства и различия с C++». Структура приложения на C# и Java. Создание классов и объектов в C# и Java. Методы-члены класса и виртуальные методы C# и Java. Механизмы наследования в C# и Java. Параметризованные типы в языках C# и Java. Определение типа объекта.	7	16	2		6	4		
17	Лекция 17. Тема: «Библиотека классов .NET Framework». Классы библиотеки. Приложения – диалоги библиотеки классов .NET Framework.	7	17	2			4		
Итого						34	34	76	

4.2. Содержание лабораторных (практических, семинарских) занятий

№ п/п	№ лекции из рабочей программы	Наименование лабораторного (практического, семинарского) занятия	Рекомендуемая литература и методические разработки (№ источника из списка литературы)	Кол-во часов
1	Лк №1,2,3	Создание классов и объектов.	№1-18	4
2	Лк №4,5,6	Разработка классов с конструкторами и деструкторами.	№1-18	4
3	Лк №7,8,9	Простое и множественное наследование.	№1-18	4
4	Лк №10,11	Перегрузка операторов.	№1-18	4
5	Лк №12	Полиморфизм и виртуальные	№1-18	4

		функции. Абстрактных классов.		
6	Лк №13,14	Шаблоны функций и классов.	№1-18	4
7	Лк №15	Разработка программ с использованием классов исключений VCL – библиотеки.	№1-18	4
8	Лк №16	Разработка классов на языке C# и Java.	№1-18	6
Итого				34

4.3. Тематика для самостоятельной работы студента

№ п/п	Тематика по содержанию дисциплины, выделенная для самостоятельного изучения	Кол-во часов из содержания дисциплины	Рекомендуемая литература и источники информации	Форма контроля СРС
1.	Составные типы данных в C++. Динамическое распределение памяти.	2	№1-18	Вх.контр
2.	ООП как направление развития теории и практики создания ПО.	4	№1-18	КР№1
3.	Понятие класса и объекта в C++.	4	№1-18	
4.	Конструкторы и деструкторы в C++.	8	№1-18	
5.	Члены класса, общие для всех экземпляров. Константные данные класса в C++.	6	№1-18	
6.	Друзья класса (friend) в C++.	8	№1-18	
7.	Механизмы наследования в C++.	8	№1-18	КР№2, КР№3
8.	Перегрузка в C++.	8	№1-18	
9.	Полиморфизм и виртуальные функции в C++.	8	№1-18	
10.	Параметризованные типы в C++.	4	№1-18	КР№3

11.	Стандартная библиотека шаблонов C++.	4	№1-18	
12.	Классы VCL-библиотеки.	4	№1-18	
13.	Объектно-ориентированное программирование на C# и Java. Сходства и различия с C++.	4	№1-18	экзамен
14.	Библиотека классов .NET Framework.	4	№1-18	экзамен
	Итого	76		

4.4. Тематика курсовых работ.

Цель курсовой работы - приобретение студентами навыков самостоятельной разработки, отладки и документирования объектно-ориентированных программ.

В задачи курсовой работы входят:

- анализ задания;
- обзор возможностей используемого языка программирования;
- разработка структуры классов;
- разработка иерархий классов;
- проектирование взаимодействий классов;
- программирование задачи;
- тестирование программы и анализ полученных результатов.

Студентам выдаются индивидуальные задания по одной из следующих тематик: программирование игр, базы данных, программы – эмуляторы, программы, работающие с графикой.

Примерные темы курсовых работ

1. Программа – эмулятор гоночной трассы (со случайными препятствиями, с 4 автомобилями)

На бесконечной трассе шириной 10 единиц двигаются 4 автомобиля (вернее движется сама трасса сверху вниз). Сверху появляются препятствия (не более 3-х общей шириной не более 6 единиц). Препятствия реализовать объектами, создавать их динамически, при выходе их за нижнюю границу удалять из памяти. Автомобили также реализовать в виде объектов (ширина автомобиля 2 единицы), автомобили могут двигаться и по горизонтали, и по вертикали, не накладываясь друг на друга. Для автомобилей предусмотреть обязательное наличие между препятствиями двух «коридоров» по 2 единицы шириной. При столкновении автомобиля с препятствием автомобиль удаляется с трассы и из памяти. Игра заканчивается, когда на трассе остается один автомобиль (о чем вывести сообщение). Всеми автомобилями управляет компьютер.

2. Программа – дизайнер комнаты (с заранее заданными элементами мебели)

3. Игра «Крестики – нолики».

4. Игра «Жизнь».

5. Разработка объектной модели «Автозаправка».

5. Образовательные технологии.

В соответствии с требованиями ФГОС ВО по направлению подготовки реализация компетентного подхода должна предусматривать широкое использование в учебном процессе активных и интерактивных форм проведения занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся.

Удельный вес занятий, проводимых в интерактивных формах, определяется главной целью программы, особенностью контингента обучающихся и содержанием дисциплины, и в целом в учебном процессе они составляют не менее 20% аудиторных занятий.

При проведении занятий по учебной дисциплине рекомендуется следовать и традиционным технологиям, в частности, в каждом разделе курса выделять наиболее важные моменты, акцентировать на них внимание обучаемых.

При чтении лекций по всем разделам программы иллюстрировать теоретический материал большим количеством примеров, что позволит сделать изложение наглядным и продемонстрировать обучаемым приемы программирования.

При изучении всех разделов программы добиться точного знания обучаемыми основных исходных понятий.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

Вопросы входного контроля для проверки знаний студентов

1. Формат программы на C++.
2. Директивы препроцессора.
3. Основные типы данных C++.
4. Объявление переменных и констант в C++.
5. Операции C++.
6. Управляющие конструкции C++.
7. Объявление функций в C++. Способы передачи параметров функциям.
8. Составные типы данных в C++: массивы, структуры и объединения.
9. Указатели в C++. Работа с массивами и структурами с помощью указателей.
10. Динамическое распределение памяти.

Контрольные вопросы для проверки текущих знаний студентов

Контрольная работа №1

1. Анализ причин сложности современных программных систем.
2. Основы структурного подхода к программированию.
3. Модульное программирование.
4. Основы объектного подхода к программированию.
5. Преимущества и недостатки объектного подхода.
6. Основные области применения ООП.

7. Преимущества инкапсуляции.
8. Суть наследования и его виды.
9. Мономорфизм и полиморфизм.
10. Понятие объекта в ООП.
11. Состояние и поведение объекта.
12. Понятие класса в ООП.
13. Основные объектно-ориентированные языки программирования.
14. Структура класса C++.
15. Уровни доступа к членам класса.
16. Объявление объектов. Создание массивов объектов.
17. Указатели на объекты класса. Динамические объекты и массивы объектов.
18. Синтаксис и семантика конструктора.
19. Свойства конструкторов.
20. Конструктор по умолчанию.
21. Конструктор с параметрами.
22. Конструктор со списком инициализаторов.
23. Конструктор копирования.
24. Синтаксис и семантика деструктора.
25. Свойства деструкторов.
26. Указатель this.
27. Встраиваемые функции.
28. Статические члены класса.
29. Константные члены класса.

Контрольная работа №2

1. Модифицируемые члены класса.
2. Дружественные функции.
3. Дружественные классы.
4. Объявление классов при простом наследовании.
5. Спецификаторы наследуемого доступа.
6. Конструкторы и деструкторы производных классов.
7. Виртуальные базовые классы.
8. Замещение членов базового класса.
9. Внутреннее и защищенное наследование.
10. Назначение перегрузки функций.
11. Перегрузка конструкторов.

Контрольная работа №3

1. Назначение перегрузки операторов.
2. Перегрузка бинарных операторов.
3. Перегрузка унарных операторов.
4. Перегрузка оператора доступа к членам класса.
5. Перегрузка и неоднозначность.
6. Виды полиморфизма.
7. Виртуальные функции.
8. Абстрактные классы.

9. Назначение виртуальных деструкторов.
10. Пространство имен.
11. Назначение шаблонных функций
12. Конкретизация шаблонной функции
13. Синтаксис шаблонного класса
14. Специализация шаблонов функций и классов
15. Достоинства и недостатки шаблонов
16. Назначение и состав библиотеки STL
17. Последовательные контейнеры
18. Контейнеры – адаптеры
19. Ассоциативные контейнеры
20. Итераторы. Обратные итераторы
21. Алгоритмы библиотеки STL
22. Иерархия классов VCL-библиотеки
23. Объекты, компоненты, элементы управления VCL-библиотеки
24. Классы командных кнопок, флажков и радиокнопок VCL-библиотеки
25. Классы текстовых компонентов, списков и панелей VCL-библиотеки
26. Классы стандартных диалогов VCL-библиотеки
27. Обработка событий средствами VCL-библиотеки

Экзаменационные вопросы

1. Причины появления и актуальность ООП
2. Преимущества и недостатки объектного подхода. Область применения ООП
3. Абстрагирование и инкапсуляция как главные элементы объектной модели
4. Модульное программирование. Модули в объектно-ориентированных программах
5. Иерархические структуры в ООП. Наследование и его виды.
6. Понятие объекта. Состояние и поведение объекта. Взаимодействие объектов
7. Понятие класса. Защита данных на уровне класса. Взаимодействие классов
8. Объектно-ориентированные языки программирования
9. Структура класса на языке C++. Уровни доступа к членам класса
10. Члены класса. Функции-члены класса, определенные в классе и вне класса
11. Объявление объектов. Указатели на объекты. Массивы объектов. Динамические объекты
12. Синтаксис и семантика конструктора. Свойства конструктора
13. Конструкторы по умолчанию и конструкторы с параметрами
14. Конструкторы копирования. Конструкторы преобразования
15. Синтаксис и семантика деструкторов. Свойства деструктора
16. Статические члены класса. Статические данные. Статические функции.
17. Константные функции-члены класса. Константные объекты.
18. Дружественные функции
19. Дружественные классы

20. Простое наследование. Метка видимости `protected`. Спецификаторы наследуемого доступа
21. Конструкторы и деструкторы производного класса
22. Множественное наследование. Виртуальные базовые классы
23. Назначение перегрузки функций. Декорирование имен функций. Ограничения при перегрузке функций
24. Перегрузка конструкторов. Перегрузка и неоднозначность
25. Перегрузка операторов. Перегрузка бинарных операторов. Перегрузка унарных операторов
26. Полиморфизм. Динамический полиморфизм. Применение динамического полиморфизма
27. Виртуальные функции. Виртуальные деструкторы. Абстрактные классы. Чистые виртуальные функции
28. Виды ошибок. Исключительные ситуации. Обработка исключительных ситуаций
29. Шаблоны функций. Параметры шаблонов. Конкретизация шаблонной функции. Перегрузка шаблонов функции
30. Шаблоны классов. Статические члены шаблонных классов. Специализация шаблонов классов
31. Назначение и состав библиотеки STL. Контейнерные классы. Типы контейнеров.
32. Последовательные контейнеры: векторы, списки, деки. Адаптеры контейнеров
33. Ассоциативные контейнеры: карты, мультикарты, множества, битовые множества
34. Итераторы. Разновидности итераторов. Обратные итераторы
35. Алгоритмы. Модифицирующие и немодифицирующие операции с последовательностями
36. Иерархия классов VCL-библиотеки
37. Объекты, компоненты, элементы управления VCL-библиотеки
38. Классы командных кнопок, флажков и радиокнопок VCL-библиотеки
39. Классы текстовых компонентов, списков и панелей VCL-библиотеки
40. Классы стандартных диалогов VCL-библиотеки
41. Обработка событий средствами VCL-библиотеки
42. Структура приложения на C# и Java
43. Создание классов и объектов C# и Java
44. Методы-члены класса и виртуальные методы C# и Java
45. Механизмы наследования C# и Java
46. Классы библиотеки .NET Framework
47. Приложения – диалоги библиотеки классов .NET Framework

Вопросы проверки остаточных знаний

1. Преимущества применения ООП при создании ПО
2. Основные концепции ООП
3. Понятие объекта
4. Понятие класса
5. Простое и множественное наследование

6. Конструкторы и деструкторы класса
7. Перегрузка функций. Перегрузка конструктора
8. Перегрузка операторов. Перегрузка бинарных и унарных операторов
9. Динамический полиморфизм
10. Абстрактные классы
11. Шаблоны классов
12. Шаблоны функций
13. Библиотека стандартных шаблонов

7. Учебно-методическое и информационное обеспечение дисциплины «Объектно-ориентированное программирование»

7.1. Рекомендуемая литература и источники информации

№	Виды занятий (лк, лб, срс)	Комплект необходимой учебной лит-ры по дисциплинам (наименование учебника, пособия)	Авторы	Издат-во и год издания	Кол-во пособий, учебников и прочей лит-ры	
					в библи	на каф
О С Н О В Н А Я						
1.	Лк, лб, срс	Основы объектно-ориентированного программирования: учебное пособие	Букунов, С. В., О. В. Букунова	СПб. : Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2017. — 196 с.	Режим доступа: http://www.iprbookshop.ru/74339.html	
2.	Лк, лб, срс	Объектно-ориентированное программирование. Часть 1: лабораторный практикум	Николаев, Е. И.	Ставрополь : Северо-Кавказский федеральный университет, 2015. — 183 с.	Режим доступа: http://www.iprbookshop.ru/62966.html	
3.	Лк, лб, срс	Объектно-ориентированное программирование. Часть 2: лабораторный практикум	Николаев, Е. И.	Ставрополь : Северо-Кавказский федеральный университет, 2015. — 156 с.	Режим доступа: http://www.iprbookshop.ru/63218.html	
4.	Лк, лб, срс	Объектно-ориентированное программирование: учебное пособие	Николаев, Е. И.	Ставрополь : Северо-Кавказский федеральный университет, 2015. — 225 с.	Режим доступа: http://www.iprbookshop.ru/62967.html	
5.	Лк, лб,	Объектно-ориентированное	Васильев, А. Н.	СПб. : Наука и Техника, 2016.	Режим доступа:	

	срс	программирование на С++		— 544 с.	http://www.iprbookshop.ru/60648.html	
6.	Лк, лб, срс	Технология разработки объектно-ориентированных программ на JAVA: учебно-методическое пособие	Васюткина, И. А.	Новосибирск : Новосибирский государственный технический университет, 2012. — 152 с.	Режим доступа: http://www.iprbookshop.ru/45047.html	
7.	Лк, лб, срс	Объектно-ориентированное программирование на языке Microsoft Visual C# в среде разработки Microsoft Visual Studio 2008 и .NET Framework. 4.3	Казанский, А. А.	А. Казанский. — Электрон. текстовые данные. — М. : Московский государственный строительный университет, ЭБС АСВ, 2011. — 180 с.	Режим доступа: http://www.iprbookshop.ru/19258.html	
8.	Лк, лб, срс	Технологии объектно-ориентированного программирования: учебное пособие для ВУЗов.	Хорев П.Б.	М.:Изд.дом «Академия», 2008	30	1
ДОПОЛНИТЕЛЬНАЯ						
9.	Лк, лб, срс	Объектно-ориентированное программирование: практикум: учеб. пособие для вузов	Павловская Т.А., Щупак Ю.А.	СПб.: Питер, 2006, 265 с.	1	-
10.	Лк, лб, срс	Как программировать на С++. 5-е издание	Дейтел Х.М., Дейтел П.Ж.	М.: Изд. «БИНОМ», 2008	1	-
11.	Лк, лб, срс	С++. Программирование на языке высокого уровня	Павловская Т.А.	СПб.: Питер, 2002, 464с.	1	-
12.	Лк, лб, срс	Основы программирования: учебник для студ.	Семакин И.Г., Шестаков А.П.	М.:Изд.центр «Академия», 2007	30	1

13.	Лк, лб, срс	Язык программирования Си++. Курс лекций. Учебное пособие/Издание второе, исправленное	Фридман А.Л.	М.: ИНТУИТ.РУ «Интернет-университет Информационных технологий», 2004, 264с.	1	-
14.	Лк, лб, срс	Язык Си++. Учебное пособие. 5 издание.	Подбельский В.В.	М.: Финансы и статистика, 2001г	15	1
15.	Лк, лб, срс	Эффективное использование STL/C	Мейерс С.	СПб.: Питер, 2002	2	1
16.	Лк, лб, срс	Программирование на Visual C++	Глушаков С.В., Коваль А.В., Смирнов С.В.	2003, 728с	1	-
17.	Лк, лб, срс	Теория и практика C++	Шилдт Г.	СПб.: ВHV-Санкт-Петербург, 1996, 416с	1	-
18.	Лк, лб, срс	Философия Java. Библиотека программиста	Эккель Б.	СПб.: Питер, 2001, 880с.	1	-

7.2. Программное обеспечение.

Интегрированные среды разработки программ Borland Developer Studio и Visual Studio . NET, базы данных, информационно – справочные и поисковые системы; вузовские электронно-библиотечные системы учебной литературы; база научно-технической информации ВИНТИ РАН.

8. Материально-техническое обеспечение дисциплины.

Для проведения лекционных занятий на факультете имеется комплект технических средств обучения в составе:

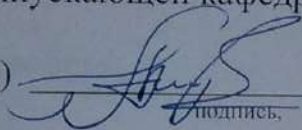
- интерактивная доска;
- переносной компьютер (в конфигурации не хуже: процессор IntelCore 2 Duo, 2 Гбайта ОЗУ, 500 Гбайт НЖМД);
- проектор (разрешение не менее 1280x1024);

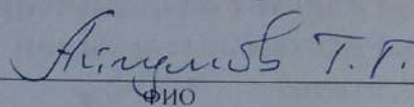
Для проведения лабораторных занятий имеется компьютерный класс, оборудованный компьютерами с установленным программным обеспечением, предусмотренным программой дисциплины.

Программа составлена в соответствии с требованиями ФГОС ВО с учетом рекомендаций и примерной ООП ВО по направлению 09.03.04 Программная инженерия и профилю подготовки «Разработка программно-информационных систем».

Рецензент от выпускающей кафедры (работодателя) по направлению

(специальности)


подпись,


ФИО