


ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«КАМЧАТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КамчатГТУ»)

Факультет информационных технологий

Кафедра «Системы управления»

УТВЕРЖДАЮ

Декан ФИТ

 И.А. Рычка

«27» 03 2020г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Объектно-ориентированное программирование»

направление подготовки:

09.03.01 «Информатика и вычислительная техника»
(уровень бакалавриата)

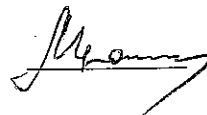
направленность (профиль):

«Программное обеспечение средств вычислительной техники и
автоматизированных систем»

Петропавловск-Камчатский
2020

Рабочая программа разработана в соответствии с ФГОС ВО по направлению подготовки 09.03.01 «Информатика и вычислительная техника», профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем», и учебного плана ФГБОУ ВО «КамчатГТУ».

Составитель рабочей программы:
Профессор кафедры СУ, доцент, д.ф.-м.н.



Марапулец Ю.В

Рабочая программа рассмотрена на заседании кафедры «Системы управления»
Протокол №8 от «27» 03 2020 года.

«27» 03 2020г.



и.о. зав. кафедрой СУ

И.А. Рыжко

1. Цели и задачи учебной дисциплины, ее место в учебном процессе

Цели и задачи изучения дисциплины

Целью дисциплины является ознакомление студентов с парадигмой и технологией объектно-ориентированного программирования (ООП) и сопутствующих технологий программирования, а так же - обучение студентов основам ООП на языке программирования высокого уровня C++.

Задачей дисциплины является дать студенту теоретические и практические знания по принципам объектно-ориентированного проектирования и разработки объектно-ориентированного программного обеспечения ЭВМ.

В результате изучения дисциплины:

- студент должен знать основные концепции объектно-ориентированного программирования, лексические и синтаксические основы объектно-ориентированного языка программирования C++, принципы построения классов и объектов, конструкторы и деструкторы, виртуальные методы и классы, абстрактные классы, принципы и виды наследования классов, шаблоны классов, обработку исключительных ситуаций, методику объектно-ориентированного анализа и проектирования..

- студент должен уметь создавать объектно-ориентированные программы с применением классов на языке C++, создавать иерархию классов на C++, использовать полиморфизм, проектировать с учетом множественного наследования, создавать шаблоны функции и классов, использовать механизм обработки исключений, использовать библиотеку потоковых классов, производить отладку программного кода в среде разработчика Visual C++

- студент должен приобрести навыки по объектно-ориентированному проектированию и разработке объектно-ориентированного программного кода в современных операционных системах.

Краткая характеристика дисциплины

Дисциплина «Объектно-ориентированное программирование» направлена для освоения следующих компетенций основной образовательной программы подготовки бакалавра по направлению подготовки 09.03.01 "Информатика и вычислительная техника" федерального государственного образовательного стандарта высшего образования:

- способностью разрабатывать алгоритмы и программы, пригодные для практического применения (ОПК-8).

- способностью к разработке алгоритмических и программных решений в области системного и прикладного программного обеспечения (ПК-1).

Планируемые результаты обучения при изучении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы, представлены в таблице.

Таблица - Планируемые результаты обучения при изучении дисциплины, соотнесенные с планируемыми результатами освоения образовательной программы

Код компетенции	Планируемые результаты освоения образовательной программы	Планируемый результат обучения по дисциплине	Код показателя освоения
ОПК-8	Обладать способностью разрабатывать алгоритмы и программы, пригодные для практического применения	Знать: <ul style="list-style-type: none"> – основные концепции объектно-ориентированного программирования, методику объектно-ориентированного анализа и проектирования; – лексические и синтаксические основы объектно-ориентированного языка программирования C++, принципы построения классов и объектов, принципы и виды наследования классов, шаблоны классов, обработку исключительных ситуаций; – принципы разработки объектно-ориентированного ПО в среде разработчика Microsoft Visual C++. 	3(ОПК-8)1
		3(ОПК-8)2	
		3(ОПК-8)3	
		Уметь: <ul style="list-style-type: none"> – спроектировать алгоритм и структуру объектно-ориентированного ПО; – уметь разрабатывать объектно-ориентированный программный код на языке C++; – уметь разрабатывать объектно-ориентированный программный код в среде разработчика Microsoft Visual C++. 	У(ОПК-8)1
		У(ОПК-8)2	
		У(ОПК-8)3	
Владеть: <ul style="list-style-type: none"> – навыками по разработке структуры объектно-ориентированного ПО; – навыками по разработке по разработке объектно-ориентированного ПО на языке C++; – навыками по разработке объектно-ориентированного ПО в среде разработчика Microsoft Visual C++. 	В(ОПК-8)1		
В(ОПК-8)2			
В(ОПК-8)3			
ПК-1	Обладать способностью к разработке	Знать: <ul style="list-style-type: none"> – принципы разработки 	

алгоритмических и программных решений в области системного и прикладного программного обеспечения	объектно-ориентированного ПО с консольным интерфейсом;	З(ПК-1)1
	– принципы разработки объектно-ориентированного ПО с оконным интерфейсом на базе функций WinAPI;	З(ПК-1)2
	– принципы разработки объектно-ориентированного ПО с оконным интерфейсом с использованием классов MFC.	З(ПК-1)3
	Уметь:	
	– проектировать и разрабатывать объектно-ориентированный программный код с консольным интерфейсом в среде Microsoft Visual C++;	У(ПК-1)1
	– проектировать и разрабатывать объектно-ориентированный программный код с оконным интерфейсом на базе функций WinAPI;	У(ПК-1)2
– проектировать и разрабатывать объектно-ориентированный программный код с оконным интерфейсом с использованием классов MFC.	У(ПК-1)3	
	Владеть:	
– навыками разработки приложений с консольным пользовательским интерфейсом в среде Microsoft Visual C++;	В(ПК-1)1	
– навыками разработки приложений с графическим пользовательским интерфейсом на основе функций WinAPI;	В(ПК-1)2	
– навыками разработки приложений с графическим пользовательским интерфейсом на основе классов MFC.	В(ПК-1)3	

Связь с предшествующими дисциплинами

Дисциплина изучается студентом после прохождения курсов «Программирование», «Структура и алгоритмы обработки данных».

Связь с последующими дисциплинами

Данная дисциплина создает знания и навыки по разработке объектно-ориентированного программного обеспечения, которые в дальнейшем используются в

дисциплине «Технология разработки программного обеспечения», а также позволяют успешно выполнить выпускную квалификационную работу.

2. Содержание дисциплины.

Тематический план дисциплины

Наименование разделов и тем	Всего часов	Аудиторные занятия	Контактная работа по видам учебных занятий			Самостоятельная работа	Формы текущего контроля	Итоговый контроль знаний по дисциплине
			Лекции	Семинары (практические занятия)	Лабораторные работы			
Тема 1: Теоретические основы объектно-ориентированного программирования	27	12	10	-	2	15	Опрос, ПЗ	
Тема 2: Основы объектно-ориентированного программирования на языке С++	64	36	18	-	18	28	Опрос, ПЗ, РЗ	
Тема 3: Принципы разработки объектно-ориентированных программ в среде разработчика Microsoft Visual С++	53	20	6	-	14	33	Опрос, ПЗ, РЗ	
Зачет, КР		-	-	-	-	-	-	20+2
Всего	144	68	34		34	76		

*ПЗ – практическое задание, РЗ – решение задач, КС – конкретная ситуация

Дисциплинарный модуль 1(7-й семестр).

Продолжительность изучения модуля 3 недели

Тема 1: Теоретические основы объектно-ориентированного программирования

Лекция 1.1 **Введение.** Предмет дисциплины и ее задачи. Роль и место задач разработки программного обеспечения в интегрированных производственных комплексах, автоматизированных системах управления техническими объектами. Структура и содержание дисциплины, ее связь с другими дисциплинами специальности. Обзор рекомендуемой литературы.

Лекция 1.2 **Основные концепции объектно-ориентированного программирования.**

Факторы, обусловившие появление и содержание концепции ООП. Основные идеи ООП: использование объекта в качестве основной компоненты программы и децентрализация управления, реализуемое представлением программы как описания взаимодействия объектов. Инкапсуляция, наследование, полиморфизм. Объектно-

ориентированный подход к разработке программ. Место и роль ООП в теории и практике разработки программных систем.

Лекция 1.3 Основные модели объектно-ориентированного программирования.

Объект как совокупность данных и набора операций. Семантика объекта. Представление данных. Классификация методов: конструкторы, деструкторы, селекторы и модификаторы. Классы объектов: назначение и семантика. Класс как абстракция совокупности объектов. Классы и абстрактные типы данных. Объекты как экземпляры классов. Основные действия с объектами: создание, инициализация, использование, уничтожение. Отношение наследования для классов. Простое и множественное наследование. Иерархия классов.

Лекция 1.4 Современные технологии разработки и тестирования программного обеспечения.

Объектно-ориентированный стиль программирования. Инструментальные средства автоматизации проектирования программных систем (CASE-средства). Графический подход к решению проблемы автоматизации разработки программного обеспечения. Требования качеству современных программных средств. Среда Visual C++ как пример современного автоматизированного средства разработки. Основные элементы управления в Visual C++ , создание и разработка проекта. Типы мастеров проектов. Debug и Release проекты.

Лабораторная работа 1.1 **Изучение основных элементов управления среды разработчика. Разработка первого проекта.** Основные элементы управления проектом. Состав программы. Включение препроцессора. Использование функции main. Ввод данных, решение математического выражения и вывод результата на экран.

Цель работы: Изучение принципов разработки программы в среде Visual C++.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №1.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 1.5 Жизненный цикл программных продуктов.

Процесс разработки программ. Основные фазы проектирования программного обеспечения. Структура жизненного цикла. Модели жизненного цикла. Этапы разработки объектно-ориентированной системы.

СРС по модулю 1. Проработка теоретического материала по следующим темам:

1. Основные концепции объектно-ориентированного программирования [1,5,7].
2. Основные модели объектно-ориентированного программирования [1,5,7].
3. Современные технологии разработки и тестирования программного обеспечения [1].
4. Жизненный цикл программных продуктов [1].

Дисциплинарный модуль 2(7-й семестр).

Продолжительность изучения модуля 9 недель

Тема 2: Основы объектно-ориентированного программирования на языке C++

Лекция 2.1 Реализация концепции объектно-ориентированного программирования в языке программирования C++.

История возникновения языка C++. Место языка C++ во внедрении и развитии средств ООП. Значение языка C++ для профессиональной подготовки программистов. Структура программы. Ввод и вывод информации.

Лабораторная работа 2.1 **Разработка программ с операторами, указателями и массивами.** Разработка программ, использующих операторы, указатели и массивы.

Цель работы: Изучение принципов разработки программы, использующей операторы, указатели и массивы.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №2.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.2 **Общий синтаксис языка C++.**

Типы данных и операции C++. Операторы языка. Указатели и адресная арифметика. Массивы и функции. Выделение и освобождение памяти. Макросы. Подстановка лексем. Включение файлов. Заголовочные файлы. Способы размещения программы в файлах.

Лабораторная работа 2.2 **Разработка программ с операциями ввода-вывода в файл, динамическим выделением памяти и внешними функциями.** Разработка программ, использующих файловый ввод-вывод, динамическое выделение памяти и внешние функции.

Цель работы: Изучение принципов разработки программ с операциями ввода-вывода в файл, динамическим выделением памяти и внешними функциями.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №3.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.3 **Структуры данных.**

Структуры и объединения. Доступ к элементам структуры. Объединения. Битовые поля структур и объединений. Динамические структуры данных: стеки, очереди, списки, бинарные деревья.

Лабораторная работа 2.3 **Разработка программы, использующей структуры.** Разработка программы, использующей структуры.

Цель работы: Изучение принципов разработки программы, использующей структуры.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №4.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.4 **Общие принципы создания классов.**

Способы реализации инкапсуляции. Определение класса. Личная и общая части определения класса. Функции-элементы класса и функции-друзья. Объекты класса.

Статические члены объектов класса. Вложенные и локальные классы. Примеры описания и использования классов.

Лабораторная работа 2.4 **Разработка программы, использующей общие принципы построения классов.** Разработка программы, использующей элементарные классы.

Цель работы: Изучение принципов разработки программы, использующей элементарные классы.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №5.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.5 **Конструкторы и деструкторы.**

Принципы создания и уничтожения объектов. Свойства конструкторов и деструкторов. Конструкторы по умолчанию, с параметрами и копирования. Выделение и уничтожение динамических объектов класса.

Лабораторная работа 2.5 **Разработка программы, использующей конструкторы и деструкторы.** Разработка программы, использующей конструкторы и деструкторы классов.

Цель работы: Изучение принципов разработки программы, использующей конструкторы и деструкторы классов.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №6.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.6 **Наследование классов.**

Базовый и производный классы. Функции-элементы и функции-друзья. Правила доступа к элементам производного класса. Иерархия классов. Одиночное и множественное наследование. Виртуальные базовые классы. Особенности доступа при множественном наследовании. Виртуальные функции. Чистые виртуальные функции и абстрактные базовые классы.

Лабораторная работа 2.6 **Разработка программы, использующей механизм наследования классов.** Разработка программы, использующей механизм наследования классов.

Цель работы: Изучение принципов разработки программы, использующей механизм наследования классов.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №7.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.7 **Шаблоны классов.**

Шаблоны классов и функций. Наследование шаблонных классов. Правила отождествления параметров шаблона. Контейнерные классы. Применение шаблонных классов для создания контейнерных классов.

Лабораторная работа 2.7 **Разработка программы, использующей шаблоны классов.** Разработка программы, использующей шаблоны классов.

Цель работы: Изучение принципов разработки программы, использующей шаблоны классов.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №8.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.8 **Потоковые классы ввода-вывода.**

Использование функций ввода-вывода языка C++. Форматы ввода-вывода. Стандартный ввод-вывод. Ввод-вывод в файл. Строковый ввод-вывод. Ошибочные состояния потоков ввода-вывода.

Лабораторная работа 2.8 **Разработка программы, использующей потоковые классы ввода-вывода в файл.** Разработка программы, осуществляющей ввод-вывод в файл с помощью потоков.

Цель работы: Изучение принципов разработки программы, осуществляющей ввод-вывод в файл с помощью потоков.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №9.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 2.9 **Обработка исключительных ситуаций.**

Анализ различных моделей обработки исключительных ситуаций. Стандартные средства контроля подтверждений. Проверка предусловий и постусловий, вычисления инвариантов. Контроль асинхронных событий. Реализация модели обработки синхронных ситуаций с завершением в C++. Возбуждение ситуации, Описание блоков с контролем и реакций на ситуации. Система классов для описания исключительных ситуаций.

Лабораторная работа 2.9 **Разработка программы, использующей технологии обработки исключительных ситуаций.** Разработка программы, осуществляющей обработку исключений.

Цель работы: Изучение принципов разработки программы, осуществляющей обработку исключений.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №10.

Содержание отчета:

1. Цель работы;
2. Исходные тексты разработанных программ;
3. Результаты работы программ (копии экранов);
4. Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

СРС по модулю 2. Проработка теоретического материала по следующим темам:

1. Реализация концепции объектно-ориентированного программирования в языке программирования C++ [1-7].
2. Общий синтаксис языка C++ [1-7].
3. Структуры данных [1-7].
4. Общие принципы создания классов [1, 3-7].
5. Конструкторы и деструкторы [1, 3-7].
6. Наследование классов [1, 3-7].
7. Шаблоны классов [1, 3-7].
8. Поточковые классы ввода-вывода [1].
9. Обработка исключительных ситуаций [1].

Дисциплинарный модуль 3(7-семестр).

Продолжительность изучения модуля 5 недель

Тема 3: Принципы разработки объектно-ориентированных программ в среде разработчика Microsoft Visual C++

Лекция 3.1 Разработка проектов на основе функций Win API.

Виды интерфейсов API. Принципы разработки проекта на основе Win API. Типы мастеров проекта. Технологии разработки консольных и оконных интерфейсов. Использование в проекте DLL.

Лекция 3.2 Разработка проектов на основе классов MFC.

Типы мастеров проекта. Изменение установок проекта. Иерархия классов MFC. Функции MFC. Обработка сообщений в MFC. Анализ классов проекта. Создание ресурсов. Включение инструмента ClassWizard. Типы диалогов. Введение элементов в диалоговое окно. Виды применяемых элементов. Управление мышью. Поддержка вывода на принтер. Запись в файл.

Лабораторная работа 3.1 **Разработка приложения на основе интерфейса SDI.** Разработка программы-прототипа текстового редактора (4 часа).

Цель работы: Изучение принципов разработки программы простого текстового редактора на основе интерфейса SDI в среде Microsoft Visual C++.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №11.

Содержание отчета:

Цель работы;

Исходные тексты разработанных программ;

Результаты работы программ (копии экранов);

Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лабораторная работа 3.2 **Включение ресурсов в приложение.** Разработка программы-прототипа с различными ресурсами (4 часа).

Цель работы: Изучение принципов включения ресурсов в разрабатываемое приложение.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №12.

Содержание отчета:

Цель работы;

Исходные тексты разработанных программ;

Результаты работы программ (копии экранов);

Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лабораторная работа 3.3 **Создание приложения на базе диалогового окна.**
 Разработка программы на базе диалогового окна (4 часа).

Цель работы: Изучение принципов создания приложения на базе диалогового окна.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №13.

Содержание отчета:

Цель работы;

Исходные тексты разработанных программ;

Результаты работы программ (копии экранов);

Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

Лекция 3.3 **Отладка программ в Visual C++.**

Структура отладчика программ. Установка метки. Окна просмотра промежуточных результатов. Пошаговое выполнение программы.

Лабораторная работа 3.4 **Отладка программ с помощью встроенных средств отладчика.** Устранение ошибок с помощью отладчика программ.

Цель работы: Изучение принципов отладки программ в Visual C++.

Работа выполняется в соответствии с методическими указаниями по выполнению лабораторных работ. Лабораторная работа №14.

Содержание отчета:

Цель работы;

Исходные тексты разработанных программ;

Результаты работы программ (копии экранов);

Выводы по проделанной работе с указанием достоинств и недостатков предложенного исходного кода.

СРС по модулю 3. Проработка теоретического материала по следующим темам:

1. Разработка проектов на основе функций WinAPI [9-11].
2. Разработка проектов на основе классов MFC [12].
3. Отладка программ в Visual C++ [1].

3 Рейтинг - план дисциплины

	Модуль 1	Модуль 2	Модуль 3	Промежуточная аттестация	Итого
Семестр 7	10	36	29	25 (зачет)	100

4 Образовательные и информационные технологии.

Занятия, проводимые в интерактивной форме составляют 25 % от аудиторных занятий.

Вид занятия	Используемые интерактивные образовательные технологии	Количество часов
Лабораторные занятия	Совместное решение проблем	10
	Методы извлечения знаний	8
Итого		18

5 Курсовой проект (работа)

Рекомендуется следующая тематика курсовых проектов:

1. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Круг и порожденного класса Кольцо.
2. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Прямоугольник и порожденного класса Прямоугольник с кругом.
3. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Треугольник и порожденного класса Четырехугольник.
4. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Вектор на плоскости и порожденного класса Вектор и точка на плоскости.
5. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Вектор на плоскости и порожденного класса Два вектора с общим началом.
6. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Вектор на плоскости и порожденного класса Два параллельных вектора на плоскости.
7. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Круг и порожденного класса Дуга окружности.
8. Реализация возможностей объектно-ориентированного подхода с использованием структуры Комплексное число и класса Произведение комплексных чисел.
9. Реализация возможностей объектно-ориентированного подхода с использованием класса Время и дочернего класса Расписание движения поездов.
10. Реализация и графическое представление возможностей механизма наследования классов с использованием базового класса Прямоугольник и порожденного класса Прямоугольник со скругленными углами.

Критерии оценки курсового проекта

Примерный перечень критериев оценки курсовой работы (проекта)	Максимальное количество баллов
Постановка проблемы. Определение целей, задач, методов решения, объекта исследования.	6
Корректное изложение смысла основных научных идей, их теоретическое обоснование и объяснение, использование навыков научного обобщения.	6
Логичность и последовательность в изложении материала.	5
Навыки планирования и управления временем при выполнении работы. Представление работы в срок.	6
Текстовая часть (соответствие стандарту, структурная упорядоченность, ссылки, цитаты, таблицы и т.д.)	10
Графическая часть (соответствие стандартам, ВНТП и т.д.)	10
Правильность расчетов. Применение физико-математического аппарата.	5
Технико-экономическое обоснование по теме курсового проекта	6

(работы)	
Выводы и предложения по модернизации, реконструкции. Обоснованность выводов.	6
Количество и степень новизны использованных литературных источников. Способность к работе с литературными источниками, Интернет-ресурсами, справочной и энциклопедической литературой, периодической литературой.	10
Презентабельность проекта (иллюстрированность, презентации с использованием ПК и т.д.)	5
Степень самостоятельности при работе над проектом (работой).	10
Выполнение специального задания.	5
Защита курсового проекта (работы)	10
Итого	100

6 Перечень планируемых результатов

Показатели, критерии оценки сформированности компетенции, шкала оценивания результатов освоения компетенций по уровням освоения.

Уровень освоения	Критерии освоения	Показатели и критерии оценки сформированности компетенции	Шкала оценивания
Продвинуты й	<i>Компетенция сформирована.</i> Демонстрируется высокий уровень самостоятельности, высокая адаптивность практического навыка	Обучаемый демонстрирует способность к полной самостоятельности (допускаются консультации с преподавателем по сопутствующим вопросам) в выборе способа решения неизвестных или нестандартных заданий в рамках учебной дисциплины с использованием знаний, умений и навыков , полученных как в ходе освоения данной учебной дисциплины, так и смежных дисциплин. 76-100 баллов.	«отлично»
Базовый	<i>Компетенция сформирована.</i> Демонстрируется достаточный уровень самостоятельности устойчивого практического навыка	Способность обучающегося продемонстрировать самостоятельное применение знаний, умений и навыков при решении заданий, аналогичных тем, которые представлял преподаватель при потенциальном формировании компетенции, подтверждает наличие сформированной компетенции, причем на более высоком уровне. 61-75 баллов.	«хорошо»
Пороговый	<i>Компетенция сформирована.</i> Демонстрируется	Если обучаемый демонстрирует самостоятельность в применении знаний, умений и навыков к решению учебных заданий в полном соответствии	«удовлетворительно»

	недостаточный уровень самостоятельности практического навыка	с образцом, данным преподавателем, по заданиям, решение которых было показано преподавателем, следует считать, что компетенция сформирована, но ее уровень недостаточно высок. 46-60 баллов	
Низкий	<i>Компетенция не сформирована</i> Демонстрируется отсутствие самостоятельности и практического навыка	Неспособность обучаемого самостоятельно продемонстрировать наличие <i>знаний</i> при решении заданий, которые были представлены преподавателем вместе с образцом их решения, отсутствие самостоятельности в применении <i>умения</i> к использованию методов освоения учебной дисциплины и неспособность самостоятельно проявить <i>навык</i> повторения решения поставленной задачи по стандартному образцу свидетельствуют об отсутствии сформированной компетенции. Менее 45 баллов.	«неудовлетворительно»

7 Оценочные средства для проведения промежуточной аттестации.

Фонд оценочных средств содержит:

1. Тестовые задания для контроля усвоения материала.
1. Перечень вопросов к промежуточной аттестации.

Примерный перечень вопросов к промежуточной аттестации (зачет)

1. Факторы, обусловившие появление и содержание концепции ООП.
2. Основные свойства ООП: абстракция, инкапсуляция, наследование, полиморфизм.
3. Основные фазы проектирования программного обеспечения. Структура жизненного цикла.
4. Модели жизненного цикла. Этапы разработки объектно-ориентированной системы.
5. История возникновения языка C++. Общий синтаксис языка C++. Разделители. Использование комментариев. Структура идентификаторов.
6. Этапы подготовки программы на языке C++ к исполнению на ПЭВМ.
7. Резервированные слова.
8. Директивы препроцессора.
9. Основные типы данных. Размер данных.
10. Определение переменных. Локальные, статические и др. типы переменных.
11. Арифметические операции. Операции присваивания.
12. Операции отношения. Логические операции. Побитовые операции.
13. Оператор-выражение. Основные виды операторов.
14. Условные операторы.
15. Операторы цикла.
16. Оператор переключения.
17. Указатели и адреса.
18. Описание массива. Массивы в качестве параметров. Строки. Массивы указателей.
19. Определение внешней функции. Вызов функции. Использование и определение параметров. Функция main.

20. Подключение библиотек. Основные библиотеки языка C++.
21. Библиотеки ввода-вывода. Основные функции ввода-вывода.
22. Ввод-вывод в файл, доступ к файлам. Форматы записи данных.
23. Математическая библиотека. Основные функции.
24. Библиотеки для работы с символами и строками. Основные функции.
25. Библиотека общего назначения. Основные функции.
26. Технологии и функции для динамического выделения памяти.
27. Системные библиотеки. Основные функции.
28. Перечисления.
29. Структуры и объединения.
30. Классы: определение, описание, члены класса. Управление доступом к элементам классов, область определения элементов.
31. Конструкторы и деструкторы. Дружественные функции класса.
32. Основы механизма наследования классов.
33. Шаблоны классов.
34. Средства обработки исключений в языке C++.

8 Учебно-методические материалы по дисциплине

Основная литература

1. *Марапулец Ю.В.* Программирование на языке высокого уровня. Рекомендовано ДВ РУМЦ в качестве учебного пособия. Петропавловск-Камчатский: Издательство КамчатГТУ, 2008 - 189 с.

Дополнительная литература

2. *Карпов Б., Баранова Т.* C++. Специальный справочник. С.-Пб.: Питер, 2001. – 479 с.
3. *Климова Л.М.* C++. Практическое программирование. М.:КУДИЦ-ОБРАЗ, 2001. – 587 с.
4. *Марапулец Ю.В.* Основы программирования на языке C++. П.-Камчатский: КамчатГТУ, 2003. – 157 с.
5. *Подбельский В.В.* Язык Си++. М.: Финансы и статистика, 1996 г. – 559 с.
6. *Романов Е.Л.* Практикум по программированию на C++. С.-Пб.: БХВ-Петербург, 2004. – 427 с.
7. *Страуструп Б.* Язык программирования Си++. М.: Радио и связь, 1991. – 352 с.
8. *Финогенов К.Г.* Win32. Основы программирования. М.: Диалог-МИФИ, 2004. - 416 с.
9. *Рихтер Д.* Windows. Создание эффективных Win32-приложений с учетом специфики 64-разрядной версии Windows. С.-Пб.: Питер, 2005. - 624 с.
10. *Д.Круглински, С.Уингоу, Д.Шефферд.* Программирование на Microsoft Visual C++ для профессионалов. С.-Пб.: Питер, 2004 г.
11. *Щупак Ю.А.* Win32 API. Эффективная разработка приложений. С.-Пб.: Питер, 2007.-572с.
12. *Черносвитов А.* Visual C++ и MFC. С.-Пб.: Питер, 2005 г.

Перечень методических указаний к проведению учебных, лабораторных занятий и самостоятельной работы студентов

13. *Марапулец Ю.В.* Объектно-ориентированное программирование. Методическое пособие к выполнению лабораторных работ по направлению подготовки бакалавров 09.03.01 «Информатика и вычислительная техника» по дисциплине «Объектно-ориентированное программирование» для студентов очной формы обучения – Петропавловск-Камчатский: КамчатГТУ, 2015. – 95 с. (электронная форма).

Перечень обучающих, контролируемых компьютерных программ

1. Microsoft Visual C++.

9 Материально-техническая база

На кафедре имеется 1 аудитория для проведения лекционных занятий, 3 компьютерных класса для проведения практических и лабораторных занятий.

10 Распределение часов по модулям и темам занятий

Распределение часов по модулям и темам занятий (очная форма обучения)

№ п/п	Наименование разделов и тем	Количество часов			
		ЛК	ПЗ	ЛЗ	СРС
Дисциплинарный модуль 1 (7-й семестр)					
1.	Введение. Предмет дисциплины и ее задачи. Роль и место задач разработки программного обеспечения в интегрированных производственных комплексах, автоматизированных системах управления техническими объектами. Структура и содержание дисциплины, ее связь с другими дисциплинами специальности. Обзор рекомендуемой литературы.	2	0	0	15
2.	Основные концепции объектно-ориентированного программирования. Факторы, обусловившие появление и содержание концепции ООП. Основные идеи ООП: использование объекта в качестве основной компоненты программы и децентрализация управления, реализуемое представлением программы как описания взаимодействия объектов. Инкапсуляция, наследование, полиморфизм. Объектно-ориентированный подход к разработке программ. Место и роль ООП в теории и практике разработки программных систем.	2	0	0	
3.	Основные модели объектно-ориентированного программирования. Объект как совокупность данных и набора операций. Семантика объекта. Представление данных. Классификация методов: конструкторы, деструкторы, селекторы и модификаторы. Классы объектов: назначение и семантика. Класс как абстракция совокупности объектов. Классы и абстрактные типы данных. Объекты как экземпляры классов. Основные действия с объектами: создание, инициализация, использование, уничтожение. Отношение наследования для классов. Простое и множественное наследование. Иерархия классов.	2	0	0	
4.	Современные технологии разработки и тестирования программного обеспечения. Объектно-ориентированный стиль программирования. Инструментальные средства автоматизации проектирования программных систем (CASE-средства). Графический подход к решению проблемы автоматизации разработки программного обеспечения. Требования качеству современных программных средств. Среда Visual	2	0	2	

	C++ как пример современного автоматизированного средства разработки. Основные элементы управления в Visual C++, создание и разработка проекта. Типы мастеров проектов. Debug и Release проекты.				
5.	Жизненный цикл программных продуктов. Процесс разработки программ. Основные фазы проектирования программного обеспечения. Структура жизненного цикла. Модели жизненного цикла. Этапы разработки объектно-ориентированной системы.	2	0	0	
Итого по модулю:		10	0	2	15
Дисциплинарный модуль 2(7-й семестр)					
1.	Реализация концепции объектно-ориентированного программирования в языке программирования C++. История возникновения языка C++. Место языка C++ во внедрении и развитии средств ООП. Значение языка C++ для профессиональной подготовки программистов. Структура программы. Ввод и вывод информации.	2	0	2	28
2.	Общий синтаксис языка C++. Типы данных и операции C++. Операторы языка. Указатели и адресная арифметика. Массивы и функции. Выделение и освобождение памяти. Макросы. Подстановка лексем. Включение файлов. Заголовочные файлы. Способы размещения программы в файлах.	2	0	2	
3.	Структуры данных. Структуры и объединения. Доступ к элементам структуры. Объединения. Битовые поля структур и объединений. Динамические структуры данных: стеки, очереди, списки, бинарные деревья.	2	0	2	
4.	Общие принципы создания классов. Способы реализации инкапсуляции. Определение класса. Личная и общая части определения класса. Функции-элементы класса и функции-друзья. Объекты класса. Статические члены объектов класса. Вложенные и локальные классы. Примеры описания и использования классов.	2	0	2	
5.	Конструкторы и деструкторы. Принципы создания и уничтожения объектов. Свойства конструкторов и деструкторов. Конструкторы по умолчанию, с параметрами и копирования. Выделение и уничтожение динамических объектов класса.	2	0	2	
6.	Наследование классов. Базовый и производный классы. Функции-элементы и функции-друзья. Правила доступа к элементам производного класса. Иерархия классов. Одиночное и множественное наследование. Виртуальные базовые классы. Особенности доступа при множественном наследовании. Виртуальные функции. Чистые виртуальные функции и абстрактные базовые классы.	2	0	2	
7.	Шаблоны классов. Шаблоны классов и функций. Наследование шаблонных классов. Правила отождествления параметров шаблона. Контейнерные классы. Применение шаблонных классов для создания контейнерных классов.	2	0	2	
8.	Потоковые классы ввода-вывода. Использование	2	0	2	

	функций ввода-вывода языка C++. Форматы ввода-вывода. Стандартный ввод-вывод. Ввод-вывод в файл. Строковый ввод-вывод. Ошибочные состояния потоков ввода-вывода.				
9.	Обработка исключительных ситуаций. Анализ различных моделей обработки исключительных ситуаций. Стандартные средства контроля подтверждений. Проверка предусловий и постусловий, вычисления инвариантов. Контроль асинхронных событий. Реализация модели обработки синхронных ситуаций с завершением в C++. Возбуждение ситуации, Описание блоков с контролем и реакций на ситуации. Система классов для описания исключительных ситуаций.	2	0	2	
Итого по модулю:		18	0	18	28
Дисциплинарный модуль 3 (7-й семестр)					
1.	Разработка проектов на основе функций Win API. Виды интерфейсов API. Принципы разработки проекта на основе Win API. Типы мастеров проекта. Технологии разработки консольных и оконных интерфейсов. Использование в проекте DLL.	2	0	0	33
2.	Разработка проектов на основе классов MFC. Типы мастеров проекта. Изменение установок проекта. Иерархия классов MFC. Функции MFC. Обработка сообщений в MFC. Анализ классов проекта. Создание ресурсов. Включение инструмента ClassWizard. Типы диалогов. Введение элементов в диалоговое окно. Виды применяемых элементов. Управление мышью. Поддержка вывода на принтер. Запись в файл.	2	0	12	
3.	Отладка программ в Visual C++. Структура отладчика программ. Установка метки. Окна просмотра промежуточных результатов. Пошаговое выполнение программы.	2	0	2	
Итого по модулю:		6	0	14	33
Итого(7-й семестр)		34	0	34	76

11 ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ В РАБОЧЕЙ ПРОГРАММЕ

Дополнения и изменения в рабочей программе за ____/____ учебный год

В рабочую программу _____
(наименование дисциплины)

для специальности (тей) _____
(номер специальности)

вносятся следующие дополнения и изменения:

Дополнения и изменения внес _____
(должность, Ф.И.О., подпись)

Рабочая программа пересмотрена и одобрена на заседании кафедры

(наименование кафедры)

« ____ » _____ 20__ г.

Заведующий кафедрой _____
(подпись)

(Ф.И.О.)

РЕЙТИНГ-ЛИСТ

по дисциплине «Объектно-ориентированное программирование»
 семестр 7 группа 17ИТб
 преподаватель: Марапулец Ю.В.

Недели обучения	Аудиторная СРС		Внеаудиторная СРС			Примечание
	Вид задания	Рейтинг	Вид задания	Срок сдачи СРС	Рейтинг	
МОДУЛЬ 1						
1-2			Самостоятельная работа №1	1-2 неделя	4	
3	Лабораторная работа 1.1	2	Самостоятельная работа №2	3 неделя	4	
ИТОГО ПО МОДУЛЮ		2	ИТОГО ПО МОДУЛЮ		8	
МОДУЛЬ 2						
4	Лабораторная работа 2.1	2	Самостоятельная работа №1	3 -4 неделя	3	
5	Лабораторная работа 2.2	2				
6	Лабораторная работа 2.3	2	Самостоятельная работа №2	5-6 неделя	3	
7	Лабораторная работа 2.4	2				
8	Лабораторная работа 2.5	2	Самостоятельная работа №3	7-8 неделя	3	
9	Лабораторная работа 2.6	2				
10	Лабораторная работа 2.7	2	Самостоятельная работа №4	9-10 неделя	3	
11	Лабораторная работа 2.8	2				
12	Лабораторная работа 2.9	2	Самостоятельная работа №5	11-12 неделя	3	
ИТОГО ПО МОДУЛЮ		18			18	
МОДУЛЬ 3						
15	Лабораторная работа 3.1	2	Самостоятельная работа №1	13 неделя	6	
16	Лабораторная работа 3.2	2	Самостоятельная работа №2	14-15 неделя	5	
17	Лабораторная работа 3.3	2	Самостоятельная работа №3	16 неделя	5	
18	Лабораторная работа 3.4	2	Самостоятельная работа №4	17 неделя	5	
ИТОГО ПО МОДУЛЮ		8			21	
ИТОГО		28			43	