

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Забайкальский государственный университет»
(ФГБОУ ВО «ЗабГУ»)

Энергетический факультет

Кафедра Информатики, вычислительной техники и прикладной математики

УТВЕРЖДАЮ:

Декан факультета

Энергетический факультет

Батухтин Андрей
Геннадьевич

«____» _____ 20____
г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1.В.09 Объектно-ориентированное программирование
на 252 часа(ов), 7 зачетных(ые) единиц(ы)

для направления подготовки (специальности) 09.03.01 - Информатика и вычислительная
техника

составлена в соответствии с ФГОС ВО, утвержденным приказом
Министерства образования и науки Российской Федерации от

«____» _____ 20____ г. №____

Профиль – Программное обеспечение вычислительной техники и автоматизированных
систем (для набора 2022)

Форма обучения: Заочная

1. Организационно-методический раздел

1.1 Цели и задачи дисциплины (модуля)

Цель изучения дисциплины:

Цель. Формирование у студента умений, навыков и знаний по объектно-ориентированному анализу, проектированию и программированию с использованием современных фреймворков для объектно-ориентированных языков общего назначения и сред разработки.

Задачи изучения дисциплины:

- Изучить основы языка С++, познакомиться с отличиями от ранее изученного языка программирования;
- Изучить основные принципы объектно-ориентированного программирования;
- Изучить реализацию этих принципов на языке С++;
- Познакомится со стандартной библиотекой языка С++;
- Изучить фреймворк на языке С++ для создания приложений с графическим интерфейсом пользователя;
- Научиться проектировать и разрабатывать объектно-ориентированные программы на С++ с учётом принципов SOLID;
- Познакомится с некоторыми объектно-ориентированными паттернами проектирования.
- Познакомится с объектно-ориентированным программированием на одном из других популярных языков программирования: Java, С#, Python.

1.2. Место дисциплины (модуля) в структуре ОП

Дисциплина Б1.В.09 Объектно-ориентированное программирование входит в состав Блока Б1, часть формируемая участниками образовательных отношений в Обязательные дисциплины. Изучается на 2 курсе в 3 и 4-м семестрах. Для успешного освоения дисциплины требуется знания и навыки полученные при изучении дисциплины Программирование. Полученные теоретические и практические навыки будут востребованы при изучении дисциплин: Структуры и алгоритмы обработки данных, Базы данных, Технологии WEB-программирования, Разработка приложений для мобильных устройств, Технология программирования, Вычислительная математика; при прохождении производственных практик.

1.3. Объем дисциплины (модуля) с указанием трудоемкости всех видов учебной работы

Общая трудоемкость дисциплины (модуля) составляет 7 зачетных(ые) единиц(ы), 252 часов.

Виды занятий	Семестр 4	Семестр 5	Всего часов
Общая трудоемкость			252
Аудиторные занятия, в т.ч.	20	18	38
Лекционные (ЛК)	10	8	18
Практические	0	0	0

(семинарские) (ПЗ, СЗ)			
Лабораторные (ЛР)	10	10	20
Самостоятельная работа студентов (СРС)	124	54	178
Форма промежуточной аттестации в семестре	Зачет	Экзамен	36
Курсовая работа (курсовой проект) (КР, КП)			

2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Планируемые результаты освоения образовательной программы		Планируемые результаты обучения по дисциплине
Код и наименование компетенции	Индикаторы достижения компетенции, формируемые в рамках дисциплины	Дескрипторы: знания, умения, навыки и (или) опыт деятельности
ОПК-2	<p>ОПК-2.1.Знать: современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.</p> <p>ОПК-2.2.Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.</p> <p>ОПК-2.3.Иметь навыки: применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.</p>	<p>Знать: современных языка программирования, иметь представление о других языках программирования, знать современные технологии и программные средства, в том числе отечественного производства (при условии, что они не уступают прочим продуктам), при решении задач профессиональной деятельности</p> <p>Уметь: выбирать современные информационные технологии, языки программирования и программные средства, в том числе отечественного производства (при условии, что они не уступают прочим продуктам), при решении задач профессиональной деятельности</p>

	<p>деятельности.</p>	<p>Владеть: современными информационными технологиями, некоторыми востребованными языками программирования и программными средствами, в том числе отечественного производства (при условии, что они не уступают прочим продуктам), при решении задач профессиональной деятельности.</p>
ОПК-8	<p>основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий.</p> <p>Уметь: применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.</p> <p>Иметь навыки: программирования, отладки и тестирования прототипов программно-технических комплексов задач.</p>	<p>Знать: Знать: востребованные языки программирования, принципы работы с файловыми базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий</p> <p>Уметь: Уметь применять языки программирования и принципы работы с файловыми базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов</p> <p>Владеть: Владеть: программированием, отладкой и тестированием (модульным, системным и интеграционным) прототипов программно-технических комплексов</p>
ПК-1	<p>ПК-1.1. Знать: методологии разработки программного обеспечения, назначение и возможности средств проектирования программного обеспечения.</p> <p>ПК-1.2. Уметь: разрабатывать функциональные и иные требования к программным и программно-аппаратным средствам, осуществлять</p>	<p>Знать: Знать: Итерационную методологию разработки программного обеспечения, назначение и возможности средств проектирования программного обеспечения, в том числе для построения модульной диаграммы и диаграммы классов</p> <p>Уметь: Уметь разрабатывать технические требования к программным и программно-</p>

<p>документирование на всех этапах проектирования и разработки, анализировать или самостоятельно разрабатывать требования к программному обеспечению; проектировать программные продукты для решения практических задач согласно разработанным требованиям; создавать программное обеспечение согласно разработанным проектам. проектам для решения практических и профессиональных задач</p> <p>ПК-1.3. Иметь навыки: разработки требований к программным продуктам; использования методов и средств проектирования программного обеспечения; создания программного обеспечения по разработанным</p>	<p>аппаратным средствам, пользоваться основными средствами документирования на всех этапах проектирования и разработки, анализировать или самостоятельно разрабатывать требования к программному обеспечению; проектировать программные продукты (в том числе строить модульную диаграмму и диаграмму классов) для решения практических задач согласно разработанным требованиям; создавать программное обеспечения согласно разработанным проектам.</p> <p>Владеть: Владеть Основными приёмами разработки требований к программным продуктам; использования методов и средств проектирования программного обеспечения (CASE средствами); методологией создания программного обеспечения (с применением парадигмы ООП) по разработанным проектам для решения практических и профессиональных задач</p>
---	--

3. Содержание дисциплины

3.1. Разделы дисциплины и виды занятий

3.1 Структура дисциплины для заочной формы обучения

Модуль	Номер раздела	Наименование раздела	Темы раздела	Всего часов	Аудиторные занятия			С Р С
					Л К	П З (С З)	Л Р	
1	1.1	Язык C++	Компиляция в командной строке. Среды разработки Qt Creator и Visual Studio.	24	4	0	4	16

			<p>Основы синтаксиса языка. Некоторые возможности стандартной библиотеки: строки, динамические массивы, консольный ввод и вывод, файлы. Обзор стандартов языка C++11 и более поздних, в том числе определение типа на этапе компиляции.</p>					
2	2.1	<p>Введение в ООП Классы в C++</p>	<p>Виды декомпозиции, абстракция, ОО декомпозиция, основные понятия ООП, UML диаграмма классов. Класс. Объект. Поля. Методы. Модификаторы доступа. Инкапсуляция. Конструктор без параметров и с параметрами. Константные методы. Статические члены классов.</p>	40	6	0	6	28
	2.2	<p>Отношения между классами Полиморфизм</p>	<p>Агрегация. Ассоциация. Правило вызова конструкторов и деструкторов. Наследование. Виды наследования. Перегрузка и переопределение методов. Множественное наследование. Интерфейсы. Виды полиморфизма. Динамический полиморфизм. Абстрактные и виртуальные методы. Определение типа на этапе выполнения. Шаблоны.</p>	50	6	0	6	38
	2.3	<p>Создание</p>	<p>Создание приложений с</p>	64	5	0	5	54

		приложений с графическим интерфейсом пользователя Продвинутое средства создания программ с ГИП	графическим интерфейсом пользователя Продвинутое средства создания программ с ГИП					
	2.4	SOLID Некоторые паттерны проектирования	SOLID Некоторые паттерны проектирования	18	3	0	3	12
Итого				196	24	0	24	148

3.2. Содержание разделов дисциплины

3.2.1. Лекционные занятия, содержание и объем в часах

Модуль	Номер раздела	Тема	Содержание	Трудоемкость (в часах)
1	1.1	Компиляция в командной строке. Среды разработки Qt Creator и Visual Studio. Основы синтаксиса языка. Некоторые возможности стандартной библиотеки: строки, динамические массивы, консольный ввод и вывод, файлы. Обзор стандартов языка C++11 и более поздних, в том числе определение	Компиляция в командной строке. Среды разработки Qt Creator и Visual Studio. Основы синтаксиса языка. Некоторые возможности стандартной библиотеки: строки, динамические массивы, консольный ввод и вывод, файлы. Обзор стандартов языка C++11 и более поздних, в том числе определение типа на этапе компиляции.	4

		типа на этапе компиляции.		
2	2.1	<p>Виды декомпозиции, абстракция, ОО декомпозиция , основные понятия ООП, UML диаграмма классов. Класс. Объект. Поля. Методы. Модификаторы доступа. Инкапсуляция . Конструктор без параметров и с параметрами. Константные методы. Статические члены классов.</p>	<p>Виды декомпозиции, абстракция, ОО декомпозиция, основные понятия ООП, UML диаграмма классов. Класс. Объект. Поля. Методы. Модификаторы доступа. Инкапсуляция. Конструктор без параметров и с параметрами. Константные методы. Статические члены классов.</p>	6
	2.2	<p>Агрегация. Ассоциация. Правило вызова конструкторов и деструкторов. Наследование. Виды наследования. Перегрузка и переопределение методов. Множественное наследование. Интерфейсы. Виды полиморфизма. Динамический</p>	<p>Агрегация. Ассоциация. Правило вызова конструкторов и деструкторов. Наследование. Виды наследования. Перегрузка и переопределение методов. Множественное наследование. Интерфейсы. Виды полиморфизма. Динамический полиморфизм. Абстрактные и виртуальные методы. Определение типа на этапе выполнения. Шаблоны.</p>	8

		<p>полиморфизм. Абстрактные и виртуальные методы. Определение типа на этапе выполнения. Шаблоны.</p>		
	2.3	<p>Структура программы с ГИП. Введение в фреймворк QT. Многооконные приложения. Классы для хранения и представления данных. Язык описания ГИП</p>	<p>Структура программы с ГИП. Введение в фреймворк QT. Многооконные приложения. Классы для хранения и представления данных. Язык описания ГИП</p>	5
	2.4	<p>SOLID Некоторые паттерны проектирования</p>	<p>SOLID Некоторые паттерны проектирования</p>	3

3.2.2. Практические занятия, содержание и объем в часах

Модуль	Номер раздела	Тема	Содержание	Трудоемкость (в часах)

3.2.3. Лабораторные занятия, содержание и объем в часах

Модуль	Номер раздела	Тема	Содержание	Трудоемкость (в часах)
1	1.1	<p>Компиляция в командной строке. Среды разработки Qt Creator и Visual Studio. Основы синтаксиса языка.</p>	<p>Компиляция в командной строке. Среды разработки Qt Creator и Visual Studio. Основы синтаксиса языка. Некоторые возможности стандартной библиотеки: строки, динамические массивы, консольный ввод и вывод, файлы. Обзор стандартов языка C++11 и более поздних, в том числе определение типа на этапе</p>	4

		<p>Некоторые возможности стандартной библиотеки: строки, динамические массивы, консольный ввод и вывод, файлы. Обзор стандартов языка C++11 и более поздних, в том числе определение типа на этапе компиляции.</p>	<p>компиляции.</p>	
2	2.1	<p>Виды декомпозиции, абстракция, ОО декомпозиция, основные понятия ООП, UML диаграмма классов. Класс. Объект. Поля. Методы. Модификаторы доступа. Инкапсуляция. Конструктор без параметров и с параметрами. Константные методы. Статические члены классов.</p>	<p>Виды декомпозиции, абстракция, ОО декомпозиция, основные понятия ООП, UML диаграмма классов. Класс. Объект. Поля. Методы. Модификаторы доступа. Инкапсуляция. Конструктор без параметров и с параметрами. Константные методы. Статические члены классов.</p>	6
	2.2	<p>Агрегация. Ассоциация. Правило вызова</p>	<p>Агрегация. Ассоциация. Правило вызова конструкторов и деструкторов. Наследование. Виды наследования. Перегрузка и</p>	8

		<p>конструкторов и деструкторов. Наследование. Виды наследования. Перегрузка и переопределение методов. Множественное наследование. Интерфейсы. Виды полиморфизма. Динамический полиморфизм. Абстрактные и виртуальные методы. Определение типа на этапе выполнения. Шаблоны.</p>	<p>переопределение методов. Множественное наследование. Интерфейсы. Виды полиморфизма. Динамический полиморфизм. Абстрактные и виртуальные методы. Определение типа на этапе выполнения. Шаблоны.</p>	
	2.3	<p>Структура программы с ГИП. Введение в фреймворк QT. Многооконные приложения. Классы для хранения и представления данных. Язык описания ГИП</p>	<p>Структура программы с ГИП. Введение в фреймворк QT. Многооконные приложения. Классы для хранения и представления данных. Язык описания ГИП</p>	5
	2.4	<p>SOLID Некоторые паттерны проектирования</p>	<p>SOLID Некоторые паттерны проектирования</p>	3

3.3. Содержание материалов, выносимых на самостоятельное изучение

Модуль	Номер раздела	Содержание материалов, выносимого на самостоятельное	Виды самостоятельной деятельности	Трудоемкость (в часах)
--------	---------------	--	-----------------------------------	------------------------

		изучение		
1	1.1	Компиляция в командной строке. Среды разработки Qt Creator и Visual Studio. Основы синтаксиса языка. Некоторые возможности стандартной библиотеки: строки, динамические массивы, консольный ввод и вывод, файлы. Обзор стандартов языка C++11 и более поздних, в том числе определение типа на этапе компиляции.	Компиляция в командной строке. Среды разработки Qt Creator и Visual Studio. Основы синтаксиса языка. Некоторые возможности стандартной библиотеки: строки, динамические массивы, консольный ввод и вывод, файлы. Обзор стандартов языка C++11 и более поздних, в том числе определение типа на этапе компиляции.	16
2	2.1	Виды декомпозиции, абстракция, OO декомпозиция, основные понятия ООП, UML диаграмма классов. Класс. Объект. Поля. Методы. Модификаторы доступа. Инкапсуляция. Конструктор без параметров и с параметрами. Константные методы. Статические члены классов.	Создание компьютерной программы, Составление конспекта	28
	2.2	Агрегация. Ассоциация. Правило вызова конструкторов и деструкторов. Наследование. Виды наследования. Перегрузка и переопределение методов. Множественное наследование. Интерфейсы. Виды полиморфизма. Динамический полиморфизм. Абстрактные и виртуальные методы. Определение типа на	Создание компьютерной программы, Составление конспекта	38

		этапе выполнения. Шаблоны.		
	2.3	Структура программы с ГИП. Введение в фреймворк QT. Многооконные приложения. Классы для хранения и представления данных. Язык описания ГИП	Создание компьютерной программы, Составление конспекта	54
	2.4	SOLID Некоторые паттерны проектирования	Создание компьютерной программы, Составление конспекта	12

4. Фонд оценочных средств для проведения текущей и промежуточной аттестации обучающихся по дисциплине

Фонд оценочных средств текущего контроля и промежуточной аттестации по итогам освоения дисциплины представлен в приложении.

[Фонд оценочных средств](#)

5. Учебно-методическое и информационное обеспечение дисциплины

5.1. Основная литература

5.1.1. Печатные издания

1.

5.1.2. Издания из ЭБС

1. 1. Огнева М.В. Программирование на языке С++: практический курс [Электронный ресурс]: учеб. пособие для бакалавриата и специалитета / М.В. Огнева, Е.В. Кудрина. – Москва: Издательство Юрайт, 2018. – 335 с. – (Серия: Бакалавр и специалист). – Режим доступа: biblio-online.ru/book/7670D7EC-AC37-4675-8EAE-DD671BC6D0E4 2. Тузовский А.Ф. Объектно-ориентированное программирование [Электронный ресурс]: учеб. пособие для прикладного бакалавриата / А.Ф. Тузовский. – Москва: Издательство Юрайт, 2017. – 206 с. – (Серия: Университеты России). – ISBN 978-5-534-00849-4. – Режим доступа: biblio-online.ru/book/BDEEFB2D-532D-4306-829E-5869F6BDA5F9

5.2. Дополнительная литература

5.2.1. Печатные издания

1.

5.2.2. Издания из ЭБС

1. Зыков С.В. Программирование. Объектно-ориентированный подход [Электронный ресурс]: учебник и практикум для академического бакалавриата / С.В. Зыков. – Москва: Издательство Юрайт, 2017. – 155 с. – (Серия: Бакалавр. Академический курс). – ISBN 978-5-534-00850-0. – Режим доступа: biblio-online.ru/book/E006A65E-B936-4856-B49E-1BA48CF1A52F

5.3. Базы данных, информационно-справочные и поисковые системы

Название	Ссылка
----------	--------

6. Перечень программного обеспечения

Программное обеспечение общего назначения: ОС Microsoft Windows, Microsoft Office, ABBYY FineReader, ESET NOD32 Smart Security Business Edition, Foxit Reader, АИБС "МегаПро".

Программное обеспечение специального назначения:

- 1) Atom
- 2) JetBrains IntelliJ IDEA
- 3) Mozilla Firefox
- 4) Qt Creator
- 5) Visual Studio

7. Материально-техническое обеспечение дисциплины

Наименование помещений для проведения учебных занятий и для самостоятельной работы обучающихся	Оснащенность специальных помещений и помещений для самостоятельной работы
Учебные аудитории для проведения занятий лекционного типа	Состав оборудования и технических средств обучения указан в паспорте аудитории, закрепленной расписанием по факультету
Учебные аудитории для проведения лабораторных занятий	
Учебные аудитории для промежуточной аттестации	
Учебные аудитории для проведения групповых и индивидуальных консультаций	Состав оборудования и технических средств обучения указан в паспорте аудитории, закрепленной расписанием по кафедре
Учебные аудитории для текущей аттестации	

8. Методические рекомендации по организации изучения дисциплины

Лекционные занятия. Рекомендуется вести краткий конспект со слайдов во время занятия. Перед занятием рекомендуется повторить пройденный материал, с использованием конспектов, слайдов лекций, рекомендованных и самостоятельно найденных источников; познакомиться с планом предстоящего занятия. Подготовить вопросы по пройденным темам. Лабораторные занятия. Цель занятий – углубление и закрепление теоретических знаний, формирование у них определенных умений и навыков. Перед занятием студент должен выполнить задания. Подготовится к защите работы, знать ответы на вопросы к заданию. Рекомендуется доработать конспект. Желательно использовать несколько источников.

Самостоятельная работа. Приводит студента к получению нового знания, упорядочению и углублению имеющихся знаний, формированию у него профессиональных навыков и умений.

Рекомендуется с самого начала освоения курса работать с рекомендуемыми источниками и предлагаемыми заданиями в форме подготовки к очередному аудиторному занятию, дорабатывать конспект во время самостоятельной работы.

При этом актуализируются имеющиеся знания, а также создается база для усвоения нового материала, возникают вопросы, ответы на которые студент получает в аудитории.

По заданиям, предлагаемым для решения на лабораторных занятиях, студент должен отчитаться до наступления сессии.

Разработчик/группа разработчиков:
Сергей Владимирович Ветров

Типовая программа утверждена

Согласована с выпускающей кафедрой
Заведующий кафедрой

_____ «___» _____ 20___ г.