

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное
учреждение высшего образования
«СЕВЕРО-ВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
имени М. К. АММОСОВА»
Институт математики и информатики
Кафедра информационных технологий

УТВЕРЖДАЮ
Директор ИМИ

_____ / В. И. Афанасьева /

« ____ » _____ 20 ____ г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.ОД.2.1 – Объектно-ориентированное программирование

для программы магистратуры
по направлению подготовки
09.04.01 – Информатика и вычислительная техника

ОДОБРЕНО

Заведующий кафедрой
разработчика

_____ / _____ /

ОДОБРЕНО

Заведующий выпускаю-
щей кафедрой ИТ

_____ / _____ /

РЕКОМЕНДОВАНО

Нормоконтроль в составе
ОП пройден

_____ / _____ /

Протокол № ____ от

« ____ » _____ 20 ____ г.

Протокол № ____ от

« ____ » _____ 20 ____ г.

Протокол № ____ от

« ____ » _____ 20 ____ г.

1. АННОТАЦИЯ

к рабочей программе дисциплины Б1.В.ОД.2.1 – Объектно-ориентированное программирование Трудоемкость 3 з. е.

1.1. Цель освоения и краткое содержание дисциплины

Целью изучения дисциплины «Объектно-ориентированное программирование» является: знакомство с понятиями и овладение практическими навыками объектно-ориентированного программирования (ООП).

Краткое содержание дисциплины. Принципы ООП. Средства ООП в Си++. UML и шаблоны ОО проектирования. Обобщенное программирование и STL. Средства ООП в Java. Средства тестирования и совместной разработки ПО..

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

Таблица 1. Перечень планируемых результатов обучения

Планируемые результаты освоения программы (содержание и коды компетенций)	Планируемые результаты обучения по дисциплине
ОК-8 : способностью к профессиональной эксплуатации современного оборудования и приборов (в соответствии с целями магистерской программы), ОПК-1 : способностью воспринимать математические, естественнонаучные, социально-экономические и профессиональные знания, умением самостоятельно приобретать, развивать и применять их для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте, ПК-18 : способностью к разработке программного обеспечения для создания трехмерных изображений, ПК-6 : пониманием существующих подходов к верификации моделей программного обеспечения (ПО)	В результате изучения дисциплины обучающийся должен: <u>знать:</u> принципы объектно-ориентированной разработки программ; <u>уметь:</u> – объяснять принципы объектно-ориентированного подхода к проектированию программного обеспечения; – читать UML-диаграммы классов, вариантов использования и последовательностей; – читать код классов чужих программ на языках Java и Си++; – распознавать простые случаи использования паттернов объектно-ориентированного проектирования; – использовать наследование для реализации отношений «является» и «имеет»; <u>владеть навыками:</u> – составления несложных UML-диаграмм классов, вариантов использования и последовательностей; – проектирования, релизации и тестирования простых объектно-ориентированных программ на Java и Си++; – использования контейнерных классов стандартной библиотеки шаблонов языка Си++.

1.3. Место дисциплины в структуре образовательной программы

Таблица 2. Содержательно-логические связи дисциплины

Индекс дисциплины	Наименование дисциплины	Коды учебных дисциплин, практик	
		на которые опирается содержание дисциплины	для которых содержание дисциплины выступает опорой
Б1.В.ОД.2.1	Объектно-ориентированное программирование	–	Б1.В.ДВ.1.1 – Распределенные объектные технологии, Б1.В.ДВ.1.2 – Объектные базы данных, Б1.В.ДВ.3.1 – Программирование на языке Питон, Б1.В.ДВ.3.2 – Разработка Java-приложений, Б1.В.ДВ.4.1 – Разработка клиент-серверных приложений, Б1.В.ДВ.5.1 – Алгоритмы цифровой обработки изображений, Б1.В.ДВ.5.2 – CASE-системы разработки ПО

1.4. Язык преподавания

Русский.

2. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Таблица 3. Выписка из учебного плана

Код и название дисциплины по учебному плану	Б1.В.ОД.2.1 – Объектно-ориентированное программирование	
Курс изучения	1	
Семестр(ы) изучения	1	
Форма промежуточной аттестации (зачет/экзамен)	зачет	
Курсовой проект / курсовая работа (указать вид работы при наличии в учебном плане), семестр выполнения		
Трудоемкость (в ЗЕТ)	3 (3)	
Трудоемкость (в часах) (сумма строк №1, 2, 3), в т. ч.:	108	
№ 1. Контактная работа обучающихся с преподавателем (КР), в часах:	Объем аудиторной работы, в часах	В т. ч. с применением ДОТ или ЭО, в часах
Объем работы (в часах) (1.1.+1.2.+1.3.)	59	
1.1. Занятия лекционного типа (лекции)	12	
1.2. Занятия семинарского типа, всего, в т.ч.:		
- семинары (практические занятия, коллоквиумы и т. п.)	–	
- лабораторные работы	42	
- практикумы		
1.3. КСР (контроль самостоятельной работы, консультации)	5	
№ 2. Самостоятельная работа обучающихся (СРС) (в часах)	49	
№ 3. Количество часов на экзамен (при наличии экзамена в учебном плане)	–	

3. Содержание дисциплины, структурированное по темам с указанием отведенного на них количества академических часов и видов учебных занятий

3.1. Распределение часов по темам и видам учебных занятий

Таблица 4

Тема	Всего часов	Контактная работа, в часах									Часы СРС
		Лекции	из них с прим-м ЭО и ДОТ	Семинары (практические занятия, коллоквиумы)	из них с прим-м ЭО и ДОТ	Лабораторные работы	из них с прим-м ЭО и ДОТ	Практикумы	из них с прим-м ЭО и ДОТ	КСР (консультации)	
Тема 1. Принципы ООП.	6	2	0	0	0	4	0	0	0	0	0
Тема 2. Средства ООП в Си++.	25	2	0	0	0	10	0	0	0	1	12
Тема 3. Обобщенное программирование и STL	17	2	0	0	0	6	0	0	0	1	8
Тема 4. Средства ООП в Java.	30	2	0	0	0	12	0	0	0	1	15
Тема 5. UML и шаблоны ОО проектирования	17	2	0	0	0	6	0	0	0	1	8
Тема 6. Средства тестирования и совместной разработки ПО	13	2	0	0	0	4	0	0	0	1	6
ВСЕГО ЧАСОВ	108	12	0	0	0	42	0	0	0	5	49

3.2. Содержание тем программы дисциплины

Тема 1. Принципы ООП.

Эволюция методологий программирования. Основные принципы объектного подхода. Наследование и отношения между объектами.

Тема 2. Средства ООП в Си++.

Пространства имен. Перегрузка. Классы. Доступ к членам класса. Наследование. Виртуальные методы. Особенности копирующих конструкторов. Особенности перегрузки операторов. Проблемы множественного наследования. Обработка исключений

Тема 3. Обобщенное программирование и STL

Шаблоны функций. Шаблоны классов. Наследование и шаблоны. Контейнеры STL. Итераторы. Алгоритмы. Поточные классы.

Тема 4. Средства ООП в Java.

Базовые типы Java, объекты, классы, пакеты. Доступ к членам класса. Интерфейсы. Наследование. Виртуальные методы. Многопоточные приложения.

Тема 5. UML и шаблоны ОО проектирования

Унифицированный язык проектирования UML. Основные виды диаграмм UML. Идея шаблона проектирования, различные подходы к выделению шаблонов. Шаблоны «Наблюдатель», «Итератор», «Одиночка», «Адаптер», «Фасад», «Фабрика».

Тема 6. Средства тестирования и совместной разработки ПО

Библиотека JUnit. Подходы к совместной разработке ПО с использованием средств управления версиями.

3.3. Формы и методы проведения занятий, применяемые учебные технологии

При проведении занятий и организации СРС используются традиционные технологии сообщающего обучения, предполагающие передачу информации в готовом виде: проведение лекционных занятий, самостоятельная работа с источниками. Предусмотрено использование активных и интерактивных форм обучения с целью формирования и развития профессиональных навыков студентов — выполнение практических работ с применением компьютерных технологий.

4. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

№	Наименование раздела (темы) дисциплины	Вид СРС	Трудоемкость (в часах)	Формы и методы контроля
1	Средства ООП в Си++.	Прохождение онлайн-курса Intermediate C++. Сдача домашнего задания	12	Предъявление веб-страницы с информацией о прохождении курса. Сдача работающего кода.
2	Обобщенное программирование и STL	Сдача домашнего задания	8	Сдача работающего приложения
3	Средства ООП в Java.	Прохождение начальных разделов онлайн-курса Software Construction in Java. Сдача домашнего задания	15	Предъявление веб-страницы с информацией о прохождении разделов онлайн-курса. Сдача работающего кода.
4	UML и шаблоны ОО проектирования	Сдача домашнего задания	8	Сдача диаграмм. Сдача работающего приложения

5	Средства тестирования и совместной разработки ПО	Сдача домашнего задания	6	Сдача тестов к приложению «Быки и коровы». Публикация исходного кода приложения «Быки и Коровы» в онлайн-репозитории
	ИТОГО		49	

5. Методические указания для обучающихся по освоению дисциплины

В связи с небольшим объемом аудиторных, особенно лекционных, часов, важное значение в освоении дисциплины имеет самостоятельная работа. Она предполагает в том числе и сдачу частей онлайн-курсов на английском языке. Это требует самостоятельности и ответственности.

В диагностическом разделе дисциплины приведены тесты по каждому модулю дисциплины, которые необходимо выполнить для закрепления теоретических знаний.

Последовательное и добросовестное изучение курса является основой для выработки практических навыков использования объектно-ориентированного подхода к программированию, который является основным средством управления сложностью во многих реальных программных проектах в областях деятельности, предполагаемых стандартом подготовки по направлению «Информатика и вычислительная техника».

Рейтинговый регламент по дисциплине

Вид выполняемой учебной работы (контролирующие мероприятия)	Количество баллов (min)	Количество баллов (max)
Посещаемость	5	10
Домашние задания, онлайн курсы	25	45
Практические занятия	10	15
Тестирование	20	30
Количество баллов для получения зачета (min-max)	60	100

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

6.1. Показатели, критерии и шкала оценивания

Коды оцениваемых компетенций	Показатель оценивания (дескриптор) (по п.1.2)	Уровни освоения	Критерий оценивания	Оценка
ОК-8, ОПК-1, ПК-18, ПК-6	<p><u>знать:</u> принципы объектно-ориентированной разработки программ;</p> <p><u>уметь:</u> объяснять принципы объектно-ориентированного подхода к проектированию программного обеспечения; читать UML-диаграммы классов, вариантов использования и последовательностей; читать код классов чужих программ на языках Java и Си++; распознавать простые случаи использования паттернов объектно-ориентированного проектирования; использовать наследование для реализации отношений «является» и «имеет»;</p> <p><u>владеть навыками:</u> составления несложных UML-диаграмм классов, вариантов использования и последовательностей; проектирования, реализации и тестирования простых объектно-ориентированных программ на Java и Си++; использования контейнерных классов стандартной библиотеки шаблонов языка Си++.</p>	освоено	<p>способен выполнять большинство задач из следующего списка: объяснять понятия абстракции, инкапсуляции, наследования и полиморфизма, а также пояснять их пользу на примерах в языках Си++ и Java; реализовывать класс по UML-диаграмме класса; читать код классов чужих программ на языках Java и Си++; применять паттерн «фабричный метод» реализовывать на языках Java и Си++ класс, наследующий от другого несложного класса для реализации отношения «является» ; реализовывать на языке Си++ класс, наследующий от нескольких классов STL для реализации отношения «имеет»; составлять UML-диаграмму вариантов использования по текстовому описанию; проектировать, релизовать и тестировать простые объектно-ориентированные программы на Java и Си++; писать программы, использующие потоковые классы, классы <code>vector</code> и <code>map</code>.</p>	зачтено

		не осво- ено	не способен выполнить три и более пунктов из вышеперечисленного	не зачтено
--	--	-----------------	---	---------------

6.2. Типовые контрольные задания (вопросы) для промежуточной аттестации

Коды оцениваемых компетенций	Оцениваемый показатель (ЗУВ)	Тема	Образец типового (тестового или практического) задания (вопроса)
ОК-8, ОПК-1	знать принципы объектно-ориентированной разработки программ	Принципы ООП	Зачем нужны абстрактные классы? а) Чтобы нельзя было создавать экземпляры данного класса б) Чтобы от класса могли наследовать только другие абстрактные классы в) Чтобы класс-наследник обязательно назывался по-другому г) Чтобы все функции в классах-наследниках были виртуальными
ПК-18	объяснять принципы объектно-ориентированного подхода к проектированию программного обеспечения	Принципы ООП; Средства ООП в Си++; Средства ООП в Java.	Объясните, как используется полиморфизм в выбранном языке, реализовав метод вычисления объема для подклассов «тетраэдр» и «параллелепипед» класса «многогранник»,
ОПК-1	знать принципы объектно-ориентированной разработки программ;	Средства ООП в Си++.	Реализуйте на Си++ класс «преподаватель» и унаследуйте от него классы «постоянный сотрудник» и «преподаватель с почасовой оплатой труда», класс «преподаватель» должен иметь виртуальную функцию расчета месячной оплаты труда <code>compensation()</code> , определенную как оклад, умноженный на коэффициент, для первого класса-наследника или число проведенных часов, умноженных на ставку почасовой оплаты, для второго.
ПК-6	проектирования, релизации и тестирования простых объектно-ориентированных программ на Java и Си++;	Средства тестирования и совместной разработки ПО	Напишите разумные модульные тесты JUnit для двух методов класса, реализованного вами в вопросе 2 (или его Java-эквивалента).

6.3. Методические материалы, определяющие процедуры оценивания

Форма промежуточной аттестации: зачет

Данный вид комплексного испытания предполагает последовательное выполнение всех форм текущего контроля, таких, как тесты, прохождение онлайн-курсов и выполнение практических заданий.

Тестирование. Данная форма контроля направлена на оценку основных теоретических знаний обучающегося по мере освоения основных разделов дисциплины.

Контрольные работы. В этой форме промежуточного контроля проверяются способности обобщенного анализа имеющихся теоретических знаний и умение пользоваться специальной литературой. Во время выполнения контрольной работы разрешается пользоваться справочной литературой

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Перечень литературы

№	Автор, название, место издания, издательство, год издания учебной литературы, вид и характеристика иных информационных ресурсов	Наличие грифа, вид грифа	НБ СВФУ, кафедральная библиотека и кол-во экземпляров	Электронные издания: точка доступа к ресурсу (наименование ЭБС, ЭБ СВФУ)
Основная литература				
1	Павловская Т. А. С/С++ . Программирование на языке высокого уровня. М.: Питер, 2013	МОН РФ	14	
Дополнительная литература				
1	Слободчикова А. А. Практикум по объектно-ориентированному анализу и проектированию с помощью языка UML. Якутск: СВФУ, 2007		14	
2	Крылов Е. В. Технология, надежность и качество программного обеспечения. Техника разработки программ. М.: Высшая школа, 2007	УМО вузов России	30	
3	Плаугер П. Дж., Ли М., Степанов А. STL — стандартная библиотека шаблонов С++ . СПб: БХВ-Петербург, 2004		6	

8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее сеть-Интернет), необходимых для освоения дисциплины

1. Справочник по языку С++ и стандартной библиотеке шаблонов STL. // The C++ Resources Network – Режим доступа: <http://cplusplus.com/>
2. Онлайн-курс Software Construction in Java. – Режим доступа: <https://www.edx.org/course/software-construction-java-mitx-6-005-1x>
3. Менеджер исходного кода Git // Git source control manager – Режим доступа: <http://git-scm.com/>
4. Онлайн-курс Microsoft DEV210-2x: Intermediate C++. – Режим доступа: <https://www.edx.org/course/intermediate-c-microsoft-dev210-2x>

9. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Для проведения лекционных занятий требуется аудитория, оборудованная доской, мультимедийным проектором с экраном. Для проведения лабораторных занятий требуется компьютерный класс с подключением к интернету.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения

10.1. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

При осуществлении образовательного процесса по дисциплине используются следующие информационные технологии:

- использование на занятиях электронных изданий (чтение лекций с использованием слайд-презентаций);
- ведение учета посещаемости и выполнения учебных заданий в системе Google Docs;
- разработка обучающимися программ на языках Си++ и Java;
- организация взаимодействия с обучающимися посредством электронной почты, специализированного образовательного форума Piazza;
- компьютерное тестирование.

10.2. Перечень программного обеспечения

При осуществлении образовательного процесса по дисциплине используются следующее программное обеспечение:

- среда разработки Microsoft Visual Studio или Code::Blocks с компилятором g++;
- среда разработки IntelliJ IDEA или NetBeans;
- Java Development Kit и библиотека JUnit;
- менеджер версий Git;
- интернет-браузер.

ЛИСТ АКТУАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Б1.В.ОД.2.1 — Объектно-ориентированное программирование

[illegible]

В таблице указывается только характер изменений (например, изменение темы, списка источников по теме или темам, средств промежуточного контроля) с указанием пунктов рабочей программы. Само содержание изменений оформляется приложением по сквозной нумерации.

Содержание

1	АННОТАЦИЯ	2
1.1	Цель освоения и краткое содержание дисциплины	2
1.2	Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы	2
1.3	Место дисциплины в структуре образовательной программы	3
1.4	Язык преподавания	3
2	Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся	4
3	Содержание дисциплины, структурированное по темам с указанием отведенного на них количества академических часов и видов учебных занятий	5
3.1	Распределение часов по темам и видам учебных занятий	5
3.2	Содержание тем программы дисциплины	5
3.3	Формы и методы проведения занятий, применяемые учебные технологии . .	6
4	Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	6
5	Методические указания для обучающихся по освоению дисциплины	7
6	Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине	8
6.1	Показатели, критерии и шкала оценивания	8
6.2	Типовые контрольные задания (вопросы) для промежуточной аттестации . .	9
6.3	Методические материалы, определяющие процедуры оценивания	10
7	Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	11
8	Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее сеть-Интернет), необходимых для освоения дисциплины	11
9	Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	12
10	Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения	12
10.1	Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине	12
10.2	Перечень программного обеспечения	12