

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Минцаев Магомед Шаваянович

Должность: Ректор

Дата подписания: 22.11.2023 12:03:52

Уникальный программный ключ:

236bcc35c296f119d6aafdc22836b21db52dbc07971a86865a5825f9fa4304cc

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
имени академика М.Д. Миллионщикова



РАБОЧАЯ ПРОГРАММА

ДИСЦИПЛИНЫ

«Объектно-ориентированное программирование»

Направление подготовки

09.03.01 Информатика и вычислительная техника

профиль

«Информатика и вычислительная техника»

Квалификация

бакалавр

1. Цели и задачи дисциплины

Цель преподавания дисциплины «Объектно-ориентированное программирование» состоит в углублении студентами, получающими квалификацию бакалавра, знаний и навыков в области создания приложений, ознакомлении с принципами объектно-ориентированного подхода к проектированию и разработке программ.

Задачами дисциплины являются: изучение теоретических основ современного объектно-ориентированного программирования (ООП) и получение практических навыков применения парадигмы ООП при разработке сложных программ.

2. Место дисциплины в структуре образовательной программы

Учебная дисциплина «Объектно-ориентированное программирование» относится к вариативной (профильной) части профессионального цикла ФГОС ВО по направлению подготовки 09.03.01 Информатика и вычислительная техника (квалификация «бакалавр»).

Для освоения дисциплины «Объектно-ориентированное программирование» студент должен обладать знаниями и умениями, приобретенными в результате освоения предшествующих дисциплин:

- информационные технологии;
- программирование.

Дисциплина «Объектно-ориентированное программирование» является предшествующей и необходимой для изучения следующих дисциплин:

- web-программирование;
- проектирование систем управления;
- моделирование компьютерных систем.

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

- способность осваивать методики использования программных средств для решения практических задач (ОПК-2);
- способность разрабатывать компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии программирования (ПК-2);
- способность обосновывать принимаемые проектные решения, осуществлять постановку и выполнять эксперименты по проверке их корректности и эффективности (ПК-3).

В результате освоения дисциплины студент должен

знать:

- современное состояние и перспективные направления развития технологий программирования;

- основные концепции объектно-ориентированного программирования;
- методы и технологии разработки математических, алгоритмических, информационных и программных средств ИТ;

- принципы построения классов и объектов, конструкторы и деструкторы;
- принципы и виды наследования классов, шаблоны классов;
- методику объектно-ориентированного анализа и проектирования;

уметь:

- использовать на практике принципы ООП;
- описывать предметную область в виде объектной модели;
- создавать иерархию классов;
- применять средства и библиотеки объектно-ориентированного программирования в собственных разработках;

владеть:

- общей подготовкой для решения практических задач в области разработки приложений;

- навыками отладки сложных программ.

4. Объем дисциплины и виды учебной работы

Таблица 1

Вид учебной работы	Всего часов / зач. ед.	Семестры	
		5	6
	ОФО	ОФО	
Контактная работа (всего)	115/3,2	51/1,5	64/1,7
В том числе:			
Лекции	33/0,9	17/0,5	16/0,4
Практические занятия	-	-	-
Семинары	-	-	-
Лабораторные работы	82/2,3	34/1	48/1,3
Самостоятельная работа (всего)	137/3,8	72/2	65/1,8
В том числе:			
Курсовая работа (проект)	-	-	-
Расчетно-графические работы	-	-	-
ИТР	-	-	-
Рефераты	-	-	-
Доклады с презентациями	54/1,5	36/1	18/0,5
<i>И (или) другие виды самостоятельной работы:</i>			
Подготовка к лабораторным работам	36/1	18/0,5	18/0,5
Подготовка к практическим занятиям	-	-	-
Подготовка к зачету	18/0,5	18/0,5	-
Подготовка к экзамену	29/0,8	-	29/0,8
Вид отчетности		зач.	экс.
Общая трудоемкость дисциплины	ВСЕГО в часах	252	252
	ВСЕГО в зач. единицах	7	7

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Таблица 2

№ п/п	Наименование раздела дисциплины по семестрам	Часы лекционных занятий	Часы лабораторных занятий	Часы практических (семинарских) занятий	Всего часов
5 семестр					
1.	Основы объектно-ориентированного программирования	17	34	-	51
6 семестр					
2.	Подходы к разработке сложных программных продуктов	16	48	-	64

5.2. Лекционные занятия

Таблица 3

№ п/п	Наименование раздела дисциплины	Содержание раздела
1.	Основы объектно-ориентированного программирования	<p>Понятия объекта и класса. Основные идеи и сферы применения объектно-ориентированного подхода.</p> <p>Основные принципы ООП: абстракция, инкапсуляция, наследование и полиморфизм.</p> <p>Правила построения объектной модели: абстрагирование, инкапсуляция, модульность, иерархичность, типизация, параллелизм, сохраняемость.</p> <p>Состояние, поведение и идентичность объекта. Понятие и виды операций над объектами.</p> <p>Способы создания объектов. Отношения между объектами – связь и агрегация.</p> <p>Виды отношений между классами.</p> <p>Способы проведения объектно-ориентированного анализа.</p>
2.	Подходы к разработке сложных программных продуктов	<p>Типизация данных в программировании.</p> <p>Введение в машинное обучение. Методы и подходы в машинном обучении.</p> <p>Предмет и цель программной инженерии как области знаний. Основные принципы разработки программных продуктов.</p> <p>Уровни требований к программным продуктам.</p> <p>Понятие и основные принципы конфигурационного управления.</p> <p>Виды тестирования программного обеспечения.</p> <p>Верификация и валидация.</p>

5.3. Лабораторные занятия

Таблица 4

№ п/п	Наименование раздела дисциплины	Наименование лабораторных работ
1.	Основы объектно-ориентированного программирования	Введение в объектно-ориентированное программирование. Описание предметной области в виде объектной модели. Console Application. Использование принципов ООП. Console Application. Создание иерархии классов. Console Application. Создание объектов и свойств.
2.	Подходы к разработке сложных программных продуктов	Form Application. Создание программ с пользовательским интерфейсом. Form Application. Использование принципов ООП. Unity. Знакомство со средой разработки и сценарной концепцией. Unity. Разработка игры.

5.4. Практические (семинарские) занятия: нет

6. Самостоятельная работа студентов по дисциплине

Способ организации самостоятельной работы: подготовка презентации с устным докладом по заданной тематике; разработка программного кода по заданию преподавателя.

5 семестр

Тематика докладов с презентациями:

1. Объектно-ориентированный язык и мышление. Новая парадигма и способ видения мира.
2. Развитие объектно-ориентированного программирования.
3. Объектные возможности языка C#.
4. Объектные возможности языка Python.
5. Smalltalk – язык объектного программирования.
6. Основы объектно-ориентированного программирования на языке Java.
7. Информация, требуемая для разработки конструктора: структура объекта, условия его создания и использования в программе.
8. Области видимости и время существования переменных. Локальные, статические и глобальные переменные.
9. Интерфейсы. Синтаксис интерфейса. Реализация интерфейса.
10. Виртуальные методы, перегрузка методов и стандартных операторов.
11. Абстрактные классы: назначение, построение, использование.
12. Множественное наследование как способ представления объекта с нескольких точек зрения.
13. Повторное использование кода.
14. Понятие рекурсии; математические рекурсивные функции; примеры рекурсивных процедур.
15. UML: диаграмма классов, диаграмма объектов и диаграмма компонентов.

6 семестр

Тематика докладов с презентациями:

1. Классификация методов проектирования программных продуктов.
2. Место и роль ООП в теории и практике разработки программных систем.
3. Объектно-ориентированное проектирование программных продуктов.

Объектно-ориентированный анализ.

4. Средства визуального создания программ и объектно-ориентированное программирование.

5. Объектный подход за пределами программирования.

6. Основные средства анализа и моделирования предметной области в языке UML.

7. Событийно-управляемое, параллельное и распределенное программирование.

8. Компонентно-ориентированный подход. Компоненты и их свойства.

9. Параллельное программирование. Потоки.

10. Сложность разработки ПО. Участники процесса разработки. Проблемы разработки ПО.

11. Методика тестирования объектно-ориентированных программ.

12. Библиотеки подпрограмм: понятие и виды. Схема вызова библиотек. Статическое и динамическое связывание.

13. Организация файловой структуры программы для эффективной разработки, отладки и модификации.

14. Поддержка и сопровождение сложного программного продукта.

15. Организация командной работы над проектом.

Примеры заданий

Провести анализ и декомпозицию предметной области – выделить классы и объекты, их отношения, свойства и методы:

- университет;
- поликлиника;
- аэропорт;
- паспортный стол;
- автомобильный завод;
- гостиница;
- почта и т.д.

Учебно-методическое обеспечение для самостоятельной работы студентов:

1. Тагайцева, С.Г. Предметно-ориентированное программирование [Электронный ресурс]: учебное пособие / С.Г. Тагайцева, Т.В. Юрченко. – Электрон. текстовые данные. – Нижний Новгород: Нижегородский государственный архитектурно-строительный университет, ЭБС АСВ, 2018. – 89 с. – Режим доступа: <http://www.iprbookshop.ru/80821.html> (ЭБС «IPRbooks»).

2. Кулямин, В.В. Технологии программирования. Компонентный подход [Электронный ресурс] / В.В. Кулямин. – 2-е изд. – Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. – 590 с. – Режим доступа: <http://www.iprbookshop.ru/73733.html> (ЭБС «IPRbooks»).

3. Калентьев, А.А. Новые технологии в программировании [Электронный ресурс]: учебное пособие / А.А. Калентьев, Д.В. Гарайс, А.Е. Горяинов. – Электрон.

текстовые данные. – Томск: Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2014. – 176 с. – Режим доступа: <http://www.iprbookshop.ru/72142.html> (ЭБС «IPRbooks»).

4. Васильев, А.Н. Python на примерах [Электронный ресурс]: практический курс по программированию / А.Н. Васильев. – 2-е изд. – Электрон. текстовые данные. – СПб.: Наука и Техника, 2017. – 432 с. – Режим доступа: <http://www.iprbookshop.ru/73043.html> (ЭБС «IPRbooks»).

7. Оценочные средства

5 семестр

Вопросы к зачету

К 1-ой рубежной аттестации:

1. Понятия объекта и класса. Основные идеи и сферы применения объектно-ориентированного подхода.

2. Основные принципы ООП: абстракция, инкапсуляция, наследование и полиморфизм.

3. Правила построения объектной модели: абстрагирование, инкапсуляция, модульность.

4. Правила построения объектной модели: иерархичность, типизация, параллелизм, сохраняемость.

Ко 2-ой рубежной аттестации:

5. Состояние, поведение и идентичность объекта. Понятие и виды операций над объектами.

6. Способы создания объектов. Отношения между объектами – связь и агрегация.

7. Виды отношений между классами.

8. Способы проведения объектно-ориентированного анализа.

6 семестр

Вопросы к экзамену

К 1-ой рубежной аттестации:

1. Типизация данных в программировании.

2. Введение в машинное обучение. Методы и подходы в машинном обучении.

3. Предмет и цель программной инженерии как области знаний. Основные принципы разработки программных продуктов.

4. Уровни требований к программным продуктам.

Ко 2-ой рубежной аттестации:

5. Понятие и основные принципы конфигурационного управления.

6. Виды тестирования программного обеспечения. Верификация и валидация.

7. Библиотеки подпрограмм: понятие и виды. Статическое и динамическое связывание.

8. Событийно-управляемое, параллельное и распределенное программирование.

Образец билета рубежной аттестации:

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Грозненский Государственный Нефтяной Технический Университет
им. акад. М.Д. Миллионщикова

Кафедра «Информационные технологии»
Дисциплина «Объектно-ориентированное программирование»
1-я рубежная аттестация

Группа:

Семестр:

Билет 1

1. Понятия объекта и класса. Основные идеи и сферы применения объектно-ориентированного подхода.
2. Правила построения объектной модели: абстрагирование, инкапсуляция, модульность.

Преподаватель _____ **Мачуева Д.А.**

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Грозненский Государственный Нефтяной Технический Университет
им. акад. М.Д. Миллионщикова

Кафедра «Информационные технологии»
Дисциплина «Объектно-ориентированное программирование»
2-я рубежная аттестация

Группа:

Семестр:

Билет 1

1. Состояние, поведение и идентичность объекта. Понятие и виды операций над объектами.
2. Способы создания объектов. Отношения между объектами – связь и агрегация.

Преподаватель _____ **Мачуева Д.А.**

Образец билета к зачету:

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Объектно-ориентированное программирование» Группа: _____ Семестр: _____	
Билет 1	
<ol style="list-style-type: none">1. Понятия объекта и класса. Основные идеи и сферы применения объектно-ориентированного подхода.2. Виды отношений между классами.	
Преподаватель _____	Мачуева Д.А.
Зав. кафедрой _____	Моисеенко Н.А.

Образец билета к экзамену:

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ Грозненский Государственный Нефтяной Технический Университет им. акад. М.Д. Миллионщикова Кафедра «Информационные технологии» Дисциплина «Объектно-ориентированное программирование» Группа: _____ Семестр: _____	
Билет 1	
<ol style="list-style-type: none">1. Типизация данных в программировании.2. Предмет и цель программной инженерии как области знаний. Основные принципы разработки программных продуктов.	
Преподаватель _____	Мачуева Д.А.
Зав. кафедрой _____	Моисеенко Н.А.

Текущий контроль

Образец типового задания для лабораторных занятий

Лабораторная работа на тему «Создание классов»

В языке Python определение класса начинается с ключевого слова class, после него следует имя класса и двоеточие.

```
1 class Elevator:
2     """ Simple elevator class """
3     # Переменная класса. Сколько людей было перевезено ВСЕМИ лифтами
4     people_lifted = 0
5
6     # Конструктор класса. Вызывается при создании экземпляра класса
7     def __init__(self, name):
8         self.name = name
```

```

8         # переменная экземпляра класса. Количество
9         людей перевезенных КОНКРЕТНЫМ лифтом
10        self.people_lifted = 0
11
12        # Метод перевозки людей
13        def lift(self):
14            print ("{} lifted someone".format(self.name))
15            # Увеличиваем количество людей перевезенных ЭТИМ лифтом
16            self.people_lifted += 1
17            # Увеличиваем количество людей перевезенных ВСЕМИ лифтами
18            Elevator.people_lifted += 1
19
20        # Метод печатающий информацию о конкретном лифте
21        def info(self):
22            print (self.name, "lifted", self.people_lifted, "people out of",
23                  Elevator.people_lifted)

```

Чтобы создать экземпляр класса, следует любой переменной присвоить значение имени класса, указав в скобках аргументы, которые принимает метод `__init__()`.

```

1 elevator_1 = Elevator("OTIS")
2 elevator_2 = Elevator("PHILLIPS")

```

Чтобы получить доступ к атрибутам и методам класса в Python, следует после объекта поставить точку и написать имя переменной или метода, которые вы хотите использовать:

```

1
2 # Везем человека в лифте под именем OTIS
3 elevator_1.lift()
4 # Везем двоих человек в лифте под именем PHILLIPS
5 elevator_2.lift()
6 elevator_2.lift()
7 # Получаем информацию по лифту под именем OTIS
8 elevator_1.info()
9 # Получаем информацию по лифту под именем PHILLIPS
10 elevator_2.info()

```

Результат:

```

Python 3.4.0 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.0 (default, Apr 11 2014, 13:05:18)
[GCC 4.8.2] on linux
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
OTIS lifted someone
PHILLIPS lifted someone
PHILLIPS lifted someone
OTIS lifted 1 people out of 3
PHILLIPS lifted 2 people out of 3
>>>
Ln: 11, Col: 4

```

8. Учебно-методическое и информационное обеспечение дисциплины

Основная литература:

1. Уйманова, Н.А. Основы объектно-ориентированного программирования [Электронный ресурс]: практикум / Н.А. Уйманова, М.Г. Таспаева. – Электрон. текстовые данные. – Оренбург: Оренбургский государственный университет, ЭБС АСВ, 2017. – 156 с. – Режим доступа: <http://www.iprbookshop.ru/78808.html> (ЭБС «IPRbooks»).

2. Мейер, Б. Объектно-ориентированное программирование и программная инженерия [Электронный ресурс] / Б. Мейер. – 3-е изд. – Электрон. текстовые данные. –

М.: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. – 285 с. – Режим доступа: <http://www.iprbookshop.ru/79706.html> (ЭБС «IPRbooks»).

3. Букунов, С.В. Основы объектно-ориентированного программирования [Электронный ресурс]: учебное пособие / С.В. Букунов, О.В. Букунова. – Электрон. текстовые данные. – СПб.: Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2017. – 196 с. – Режим доступа: <http://www.iprbookshop.ru/74339.html> (ЭБС «IPRbooks»).

4. Комлев, Н.Ю. Объектно Ориентированное Программирование. Хорошая книга для Хороших Людей [Электронный ресурс] / Н.Ю. Комлев. – Электрон. текстовые данные. – М.: СОЛОН-ПРЕСС, 2014. – 298 с. – Режим доступа: <http://www.iprbookshop.ru/26923.html> (ЭБС «IPRbooks»).

Дополнительная литература:

5. Тагайцева, С.Г. Предметно-ориентированное программирование [Электронный ресурс]: учебное пособие / С.Г. Тагайцева, Т.В. Юрченко. – Электрон. текстовые данные. – Нижний Новгород: Нижегородский государственный архитектурно-строительный университет, ЭБС АСВ, 2018. – 89 с. – Режим доступа: <http://www.iprbookshop.ru/80821.html> (ЭБС «IPRbooks»).

6. Кулямин, В.В. Технологии программирования. Компонентный подход [Электронный ресурс] / В.В. Кулямин. – 2-е изд. – Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. – 590 с. – Режим доступа: <http://www.iprbookshop.ru/73733.html> (ЭБС «IPRbooks»).

7. Калентьев, А.А. Новые технологии в программировании [Электронный ресурс]: учебное пособие / А.А. Калентьев, Д.В. Гарайс, А.Е. Горяинов. – Электрон. текстовые данные. – Томск: Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2014. – 176 с. – Режим доступа: <http://www.iprbookshop.ru/72142.html> (ЭБС «IPRbooks»).

8. Васильев, А.Н. Python на примерах [Электронный ресурс]: практический курс по программированию / А.Н. Васильев. – 2-е изд. – Электрон. текстовые данные. – СПб.: Наука и Техника, 2017. – 432 с. – Режим доступа: <http://www.iprbookshop.ru/73043.html> (ЭБС «IPRbooks»).

9. Материально-техническое обеспечение дисциплины

Перечень материально-технических средств учебной аудитории для проведения занятий по дисциплине:

- учебная аудитория, доска;
- стационарные компьютеры;
- мультимедийный проектор;
- настенный экран.

Составитель:

Старший преподаватель кафедры
«Информационные технологии»



/ Мачуева Д.А. /

СОГЛАСОВАНО:

Зав. кафедрой
«Информационные технологии»



/ Моисеенко Н.А. /

Зав. выпускающей кафедрой
«Информатика и вычислительная техника»



/ Алисултанова Э.Д. /

Директор ДУМР



/ Магомаева М.А. /