

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ЭКОНОМИКИ И УПРАВЛЕНИЯ

УТВЕРЖДАЮ
Декан факультета
Экономики и
Управления Володин В.М.
« 12 » *Володин* 2016 г.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.2.6 Объектно-ориентированный анализ и программирование
(индекс дисциплины по учебному плану, наименование дисциплины)

Направление подготовки 38.03.05 "Бизнес-информатика"
(код, наименование направления подготовки)

Квалификация (степень) выпускника – *Бакалавр*

Форма обучения очная / заочная
(очная, заочная, очно-заочная)

Пенза, 2016

1. Цели освоения учебной дисциплины

Целями освоения учебной дисциплины «Б1.2.6 Объектно-ориентированный анализ и программирование» являются:

- получение студентами теоретических знаний и практических навыков в программировании на основе процедурно-ориентированного и объектно-ориентированного подхода, позволяющих решать задачи обработки числовой и символьной информации в рамках прикладных и системных задач;
- формирование у студентов профессиональных компетенций, связанных с использованием теоретических знаний в области объектного подхода к проектированию программных систем, и практических навыков в области объектно-ориентированного программирования, позволяющих творчески применять их для решения задач разработки программного обеспечения и обработки информации как в своей профессиональной деятельности, так и при выполнении курсовых и практических работ при последующем обучении на старших курсах.

2. Место учебной дисциплины в структуре ОПОП бакалавриата

Дисциплина «Объектно-ориентированный анализ и программирование» относится к обязательным дисциплинам вариативной части блок Б1 дисциплин подготовки студентов по направлению 38.03.05 «Бизнес-информатика».

Изучение дисциплины «Объектно-ориентированный анализ и программирование» базируется на положениях следующих дисциплин: «Теоретические основы информатики», «Программирование».

Основные положения данной дисциплины могут быть использованы в дальнейшем при изучении следующих дисциплин: «Распределенные системы», «Базы данных», «Функциональное программирование и интеллектуальные системы», «Хранилища данных и знаний» а также при работе над выпускной квалификационной работой.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины Б1.2.6 «Объектно-ориентированный анализ и программирование»

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению:

Коды компетенции	Наименование компетенции	Структурные элементы компетенции (в результате освоения дисциплины обучающийся должен знать, уметь, владеть)
1	2	3
ОПК-1	Способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	Знать: - задачи профессиональной деятельности, информационную и библиографическую культуру; - основные требования информационной безопасности Уметь: - решать стандартные задачи профессиональной деятельности Владеть: - информационно-коммуникационными технологиями для решения стандартных задач профессиональной деятельности

ПК-3	Способность выбирать рациональные информационные системы и информационно-коммуникативные технологии и решения для управления бизнесом	Знать: - <i>основные информационные системы и информационно-коммуникативные технологии</i>
		Уметь: - <i>выбирать наилучшие решения в сфере информационных систем и информационно-коммуникативных технологий</i>
		Владеть: - <i>основными информационными системами и информационно-коммуникативными технологиями применительно к поставленной задаче управления бизнесом</i>

4. Структура и содержание дисциплины Б1.2.6 «Объектно-ориентированный анализ и программирование»

4.1. Структура дисциплины

Общая трудоемкость освоения дисциплины составляет 3 зачетных единицы, 108 часов. Программой дисциплины предусмотрены: лекционные занятия (17 часов), лабораторные работы (34 часа), самостоятельная работа студента (57 часов) в пятом семестре.

Очная форма обучения

№ п/п	Наименование разделов и тем дисциплины	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)								Формы текущего контроля успеваемости (по неделям семестра)								
				Аудиторная работа				Самостоятельная работа				Собеседование	Коллоквиум	Проверка тестов	Проверка контр. работ	Проверка реферата	Практико-ориентированные задания для проверки умений и навыков	курсовая работа (проект)	Тестирование	
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Реферат, эссе и др.	Курсовая работа (проект)									Подготовка к экзамену
1.	Введение. Объектно-ориентированный подход к построению программного обеспечения.	5	1-2	2	2			3	3				2					2		
2.	Классификация языков и инструментальных средств объектно-ориентированного проектирования и программирования	5	3-4	2	2			3	3				4					4		
3.	Абстракция. Инкапсуляция, наследование, полиморфизм.	5	5-8	16	4		12	5	5				8					8		
4.	Основы объектно-ориентированного анализа и проектирования.	5	9-11	16	4		12	5	5				11					11		
5.	Многократное использование	5	12-14	6	2		4	3	3				14					14		

	проектов. Шаблоны проектирования																	
6.	Модульное тестирование объектно-ориентированного программного обеспечения.	5	15	6	2		4	1	1				15					15
7.	Основы документирования объектно-ориентированного программного обеспечения	5	16-17	3	1		2	1	1				17					17
8.	Подготовка к экзамену							36					36					
	Общая трудоемкость, в часах	108		51	17		34	57	21			36	Промежуточная аттестация					
Форма													Семестр					
Экзамен													5					

Заочная форма обучения

№ п/п	Наименование разделов и тем дисциплины	Семестр	Недели семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)							Формы текущего контроля успеваемости								
				Аудиторная работа				Самостоятельная работа			Собеседование	Коллоквиум	Проверка тестов	Проверка контр. работ	Проверка реферата	Практико-ориентированные задания для проверки умений и навыков	курсовая работа (проект)	Тестирование	
				Всего	Лекция	Практические занятия	Лабораторные занятия	Всего	Подготовка к аудиторным занятиям	Подготовка и написание контрольной работы									Подготовка к экзамену
1.	Введение. Объектно-ориентированный подход к построению программного обеспечения.	5		1	1			8	6	2		+			+		+		
2.	Классификация языков и инструментальных средств объектно-ориентированного	5		1	1			8	6	2		+			+		+		

	проектирования и программирования																		
3.	Абстракция. Инкапсуляция, наследование, полиморфизм.	5		8	2		6	10	8	2		+			+		+		
4.	Основы объектно-ориентированного анализа и проектирования.	5		3	1		2	8	6	2		+			+		+		
5.	Многократное использование проектов. Шаблоны проектирования	5		3	1		2	8	6	2		+			+		+		
6.	Модульное тестирование объектно-ориентированного программного обеспечения.	5		1	1			6	5	1		+			+		+		
7.	Основы документирования объектно-ориентированного программного обеспечения	5		1	1			6	5	1		+			+		+		
8.	Подготовка к экзамену							36											
	Общая трудоемкость, в часах	108		18	8		10	90	42	12	36	Промежуточная аттестация							
Форма												Семестр							
Экзамен												5							

4.2. Содержание дисциплины

Тема 1. Введение. Объектно-ориентированный подход к построению программного обеспечения.

Объектно-ориентированное мышление. Принципы объектно-ориентированного подхода. Объектно-ориентированное программирование в историческом контексте. Этапы методологии: предпроектное исследование; анализ; проектирование; реализация; тестирование; внедрение; опытная эксплуатация.

Тема 2. Классификация языков и инструментальных средств объектно-ориентированного проектирования и программирования

Важнейшие подходы к программированию: структурный/модульный, функциональный/логический, объектно-ориентированный. Особенности, преимущества и недостатки подходов к программированию. Языки и программные среды, реализующие различные парадигмы программирования. Понятийный аппарат объектно-ориентированного подхода и их математические аналоги: классы (множества), объекты, атрибуты (характеристики), методы (функции).

Тема 3. Абстракция. Инкапсуляция, наследование, полиморфизм.

Понятие абстракции в математике и программировании. Математическая модель абстракции. Степень абстракции как средство оптимизации модели предметной области. Примеры использования абстракции. Демонстрация и анализ концепции абстракции на примере разработки программ, а также при разработке классов.

Связь инкапсуляции с модульной декомпозицией программного продукта. Разделение интерфейса и реализации инкапсулированных программных объектов. Признаки инкапсуляции: общедоступный интерфейс и сокрытие реализации. Демонстрация и анализ концепции инкапсуляции при разработке классов.

Наследование. Понятие базового и производного классов. Концепции абстрактного класса и метода. Наследование реализации, поведения и свойств. Переопределение метода. Виды наследования. Единственность и множественность наследования. Особенности реализации концепции наследования в языках программирования.

Полиморфизм. Представление вариативного выбора поведения программного кода для полиморфных классов и методов. Связь концепции полиморфизма с инкапсуляцией и наследованием. Расширения полиморфизма: параметрический полиморфизм, переопределение и перегрузка методов. Демонстрация и анализ концепции полиморфизма при разработке классов.

Тема 4. Основы объектно-ориентированного анализа и проектирования.

Модели жизненного цикла программного обеспечения. Преимущества и недостатки, требования к разработчикам. Объектно-ориентированный анализ (ООА) как подход к осмыслению разрабатываемого проекта. Последовательность ООА. Концептуальная модель предметной области. Проблемы ООА, пути и средства их преодоления. Демонстрация и анализ фазы объектно-ориентированного анализа при разработке классов.

Объектно-ориентированное проектирование. Выделение первичных классов/объектов; определение и уточнение назначений объектов при помощи техники извлечения существительных; детализация отношений между объектами; построение взаимосвязей классов/объектов, детализированных до сигнатур. Описание структуры, отношений и взаимодействия объектов.

Демонстрация и анализ особенностей и проблем фазы проектирования при разработке классов.

Тема 5. Многократное использование проектов. Шаблоны проектирования

Многократное использование. Экономика повторного использования кода – теория и практика, проблемы и средства их решения. Применение шаблонов проектирования. Элементы шаблонов проектирования. Демонстрация и анализ повторного использования элементов объектно-ориентированных программных проектов при разработке классов.

Тема 6. Модульное тестирование объектно-ориентированного программного обеспечения.

Классификация видов тестирования. Модульное, сборочное и приемочное тестирование. Важность этапа реализации для обеспечения качества, надежности и предсказуемости объектно-ориентированного программного продукта.

Тема 7. Основы документирования объектно-ориентированного программного обеспечения

Создание документации. Этапы жизненного цикла программы и их связь с документацией. Состав проектной документации. Демонстрация и анализ процессов документирования при разработке классов.

1. Образовательные технологии

Технологии получения студентами теоретических знаний связаны с лекционными занятиями и основаны на принципах самостоятельной внеаудиторной работы с рекомендуемым учебно-методическим и информационным обеспечением учебной дисциплины. Освоение теоретического и методологического материала предполагается также за счет проведения Интернет-тестирования и Интернет-тренингов с использованием баз ФЕПО и специализированных сайтов.

Технология формирования и закрепления практических навыков и умений у студентов базируется на выполнении и защите лабораторных работ, основанных на компьютерных симуляциях, связанных с созданием и разработкой программ на языке программирования высокого уровня.

В целях реализации индивидуального подхода к обучению студентов, осуществляющих учебный процесс по собственной траектории в рамках индивидуального рабочего плана, изучение данной дисциплины базируется на следующих возможностях: обеспечение внеаудиторной работы со студентами в том числе в электронной образовательной среде с использованием соответствующего программного оборудования, дистанционных форм обучения, возможностей интернет-ресурсов, индивидуальных консультаций и т.д.

Обучение лиц с ограниченными возможностями здоровья проводится в зависимости от их индивидуальных потребностей. При необходимости обучающимся с ограниченными возможностями здоровья предоставляется социально-психологическая помощь и сопровождение. Обучающиеся из числа лиц с ограниченными возможностями здоровья обеспечиваются печатными и электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

2. Учебно-методическое обеспечение самостоятельной работы студентов.

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Контрольные вопросы:

1. В чем состоят недостатки процедурного и модульного программирования?
2. Что такое объектно-ориентированное программирование?
3. Каковы преимущества и цели объектно-ориентированного программирования?
4. Дайте определения следующих терминов: класс, объект, поведение.
5. Каким образом объекты обмениваются информацией?
6. Что такое конструктор?
7. Какие подходы к программированию Вам известны?
8. Каковы основные понятия объектно-ориентированного подхода к программированию?
9. Что такое класс?
10. Что такое объект?
11. Что такое атрибут класса?
12. Что такое метод?
13. Что отличает объектно-ориентированный подход к программированию от компонентно-ориентированного?
14. В чем Вы видите преимущества языков объектно-ориентированного программирования?
15. В чем состоят особенности объектно-ориентированного проектирования?
16. Почему при объектно-ориентированном проектировании необходимо

использовать инструментальные средства?

17. В чем Вы видите преимущества инструментального средства Rational Rose для объектно-ориентированного проектирования?

18. В чем состоят особенности языков программирования на основе функционального подхода?

19. В чем состоят особенности языков объектно-ориентированного программирования?

20. В чем состоят особенности языков логического программирования?

21. Какие языки программирования основаны на структурном подходе?

22. Какие цели объектно-ориентированного программирования реализует абстракция?

23. Дайте определение понятию «абстракция» в программировании.

24. Какие математические теории, формализующие понятие абстракции, Вам известны?

25. Приведите пример применения абстракции.

26. Почему для достижения эффективной абстракции важно четко определить степень детализации предметной области?

27. В чем состоит особенность абстрактных типов данных?

28. В чем Вы видите недостатки абстракции?

29. Возможно ли применение в объектно-ориентированной программе различных степеней абстракции? Приведите пример.

30. Какие цели объектно-ориентированного программирования реализует инкапсуляция?

31. Дайте определение понятию «инкапсуляция»

32. Приведите пример применения инкапсуляции.

33. Дайте определение понятию «реализация».

34. Дайте определение понятию «интерфейс».

35. В чем состоит различие между интерфейсом и реализацией?

36. Почему для достижения эффективной инкапсуляции важно четко распределить ответственность?

37. Что такое тип?

38. В чем состоит особенность инкапсулированных типов данных?

39. В чем Вы видите недостатки инкапсуляции?

40. Что такое наследование?

41. Что такое единичное (простое) наследование?

42. Что такое множественное наследование?

43. Что такое базовый класс?

44. Что такое производный класс?

45. В чем состоят ограничения на использование множественного наследования?

46. Какие элементы класса может наследовать другой класс?

47. В чем основное различие между базовым классом и производным классом?

48. В чем состоит проблема «хрупких» базовых классов?

49. Что такое полиморфизм?

50. Какими средствами реализуется полиморфизм?

51. Что такое отложенное (позднее) связывание?

52. Что такое раннее связывание?

53. Каковы преимущества и недостатки применения полиморфизма времени выполнения?

54. Каковы преимущества и недостатки применения полиморфизма времени компиляции?
55. Что такое параметрический полиморфизм?
56. Что такое переопределение метода? В каких случаях оно применяется?
57. Что такое перегрузка метода?
58. Что такое «ленивые» вычисления?
59. В чем преимущества и недостатки «ленивых» вычислений?
60. Какие трудности проведения объектно-ориентированного анализа Вам известны?
61. Что такое технология разработки программного обеспечения?
62. Что такое итеративная технология проектирования?
63. Как итеративная технология проектирования соотносится со спиральной моделью?
64. В чем состоит результат объектно-ориентированного анализа?
65. Что описывают требования к системе?
66. В чем важность повторного использования?
67. Какие элементы проекта могут быть повторно использованы?
68. Что такое шаблон проектирования?
69. Какие элементы входят в шаблон проектирования?
70. Как правильно выбрать шаблон проектирования?
71. Гарантирует ли использование шаблонов высокую долю повторного использования?
72. В чем состоит важность дисциплины проекта для повышения доли повторного использования?
73. Как связан выбор модели жизненного цикла с повторным использованием?

Задания для самостоятельной работы:

См. «Методические указания по проведению самостоятельной работы по дисциплине».

6.1 План самостоятельной работы студентов очной формы обучения

№ нед.	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
1-2	Введение. Объектно-ориентированный подход к построению программного обеспечения.	Подготовка к аудиторным занятиям	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	3
3-4	Классификация языков и инструментальных средств объектно-ориентированного проектирования и программирования	Подготовка к аудиторным занятиям	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	3
5-8	Абстракция. Инкапсуляция, наследование, полиморфизм.	Подготовка к аудиторным занятиям	Изучение рекомендуемой литературы и необходимых сайтов, решение	См. список основной и дополнительной литературы	5

			контрольных вопросов по лабораторным работам		
9-11	Основы объектно-ориентированного анализа и проектирования.	Подготовка к аудиторным занятиям	1. Изучение рекомендуемой литературы и сайтов 2. Решение контрольных вопросов по лабораторным работам	См. список основной и дополнительной литературы	5
12-14	Многokrатное использование проектов. Шаблоны проектирования	Подготовка к аудиторным занятиям	Изучение рекомендуемой литературы и сайтов	См. список основной и дополнительной литературы	3
15	Модульное тестирование объектно-ориентированного программного обеспечения.	Подготовка к аудиторным занятиям	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	1
16-18	Основы документирования объектно-ориентированного программного обеспечения	Подготовка к аудиторным занятиям	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	1
	Все темы	Подготовка к экзамену	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	36

План самостоятельной работы студентов заочной формы обучения

№ п/п	Тема	Вид самостоятельной работы	Задание	Рекомендуемая литература	Количество часов
1.	Введение. Объектно-ориентированный подход к построению программного обеспечения.	Подготовка к аудиторным занятиям, подготовка и написание контрольной работы	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	8
2.	Классификация языков и инструментальных средств объектно-ориентированного проектирования и программирования	Подготовка к аудиторным занятиям, подготовка и написание контрольной работы	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	8
3.	Абстракция. Инкапсуляция, наследование, полиморфизм.	Подготовка к аудиторным занятиям, подготовка и написание контрольной работы	Изучение рекомендуемой литературы и необходимых сайтов, решение	См. список основной и дополнительной литературы	10

			контрольных вопросов по лабораторным работам		
4.	Основы объектно-ориентированного анализа и проектирования.	Подготовка к аудиторным занятиям, подготовка и написание контрольной работы	1. Изучение рекомендуемой литературы и сайтов 2. Решение контрольных вопросов по лабораторным работам	См. список основной и дополнительной литературы	8
5.	Многочисленное использование проектов. Шаблоны проектирования	Подготовка к аудиторным занятиям, подготовка и написание контрольной работы	Изучение рекомендуемой литературы и сайтов	См. список основной и дополнительной литературы	8
6.	Модульное тестирование объектно-ориентированного программного обеспечения.	Подготовка к аудиторным занятиям, подготовка и написание контрольной работы	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	6
7.	Основы документирования объектно-ориентированного программного обеспечения	Подготовка к аудиторным занятиям, подготовка и написание контрольной работы	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	6
8.	Все темы	Подготовка к экзамену	Изучение рекомендуемой литературы	См. список основной и дополнительной литературы	36

6.2. Методические указания по организации самостоятельной работы студентов

1. Подготовка к аудиторным занятиям представляет собой изучение рекомендуемой литературы и сайтов, решение контрольных вопросов по лабораторным работам.

2. Подготовка к экзамену представляет собой умение обобщить основные теоретические, методические и прикладные положения по данной дисциплине, а также опыт, полученный при написании лабораторных работ, ответов на вопросы собеседований (для усвоения знаний по дисциплине) и при решении практико-ориентированных заданий (для получения умений и навыков).

6.3. Материалы для проведения текущего и промежуточного контроля знаний студентов

№ п\п	Вид контроля	Контролируемые темы (разделы)	Компетенции, компоненты которых контролируются
1.	Собеседование 1, решение практико-ориентированных заданий для проверки умений и навыков	Введение. Объектно-ориентированный подход к построению программного обеспечения.	ОПК-1, ПК-3
2.	Собеседование 2,	Классификация языков и	ОПК-1, ПК-3

	решение практико-ориентированных заданий для проверки умений и навыков	инструментальных средств объектно-ориентированного проектирования и программирования	
3.	Собеседование 3, решение практико-ориентированных заданий для проверки умений и навыков	Абстракция. Инкапсуляция, наследование, полиморфизм.	ОПК-1, ПК-3
4.	Собеседование 4, решение практико-ориентированных заданий для проверки умений и навыков	Основы объектно-ориентированного анализа и проектирования.	ОПК-1, ПК-3
5.	Собеседование 5, решение практико-ориентированных заданий для проверки умений и навыков	Множественное использование проектов. Шаблоны проектирования	ОПК-1, ПК-3
6.	Собеседование 6, решение практико-ориентированных заданий для проверки умений и навыков	Модульное тестирование объектно-ориентированного программного обеспечения.	ОПК-1, ПК-3
7.	Собеседование 7, решение практико-ориентированных заданий для проверки умений и навыков	Основы документирования объектно-ориентированного программного обеспечения	ОПК-1, ПК-3
8.	Проверка контрольных работ (для заочной формы обучения)	Все изучаемые темы	ОПК-1, ПК-3
9.	Подготовка к экзамену	Все изучаемые темы	ОПК-1, ПК-3

Текущий контроль успеваемости в виде контрольных точек проводится по результатам защиты 4-х лабораторных работ в пятом семестре, каждая из которых оценивается по 9-15 баллов. Количество контрольных точек и сроки их проведения в семестре устанавливается по решению деканата.

Темы лабораторных работ

1. Проектирование классов
2. Создание простого класса
3. Перегрузка операторов
4. Наследование
5. Создание класса для работы с файлами

По результатам текущего контроля студент может получить от 0 до 60 баллов. Для допуска к экзамену и защите курсовой работы студент должен набрать не менее 36 баллов.

Собеседование 1. Введение. Объектно-ориентированный подход к построению программного обеспечения.

- Дайте определение объекта.
- Что такое класс?

- Какие функции выполняет конструктор класса?
- Может ли класс иметь несколько конструкторов?
- Может ли класс иметь несколько деструкторов?

Практико-ориентированные задания для проверки умений и навыков по теме 1

Изобразите графически возможное содержимое объекта «квадратное уравнение» (поля и методы).

Изобразите иерархическую схему наследования. Изобразите схему взаимодействия объектов при множественном наследовании.

Собеседование 2. Классификация языков и инструментальных средств объектно-ориентированного проектирования и программирования.

- Сформулируйте основные преимущества объектно-ориентированного подхода к программированию.
- Каковы причины развития объектно-ориентированного подхода?
- В чем состоят особенности языков программирования на основе функционального подхода?
- В чем состоят особенности языков объектно-ориентированного программирования?
- В чем состоят особенности языков логического программирования?
- Какие языки программирования основаны на структурном подходе?

Практико-ориентированные задания для проверки умений и навыков по теме 2

Приведите синтаксис описания класса.

Опишите (синтаксически) класс для работы с графической точкой (укажите все поля и методы).

Собеседование 3. Абстракция. Инкапсуляция, наследование, полиморфизм.

- Сформулируйте принцип наследования.
- Сформулируйте принцип полиморфизма.
- Сформулируйте принцип инкапсуляции.
- Каким атрибутом доступа должен быть описан элемент класса, чтобы его можно было использовать только внутри класса?
- Сколько атрибутов доступа к элементам класса имеется в Си++?

Практико-ориентированные задания для проверки умений и навыков по теме 3

Приведите синтаксис любого класса, в котором отображалось бы свойство инкапсуляции.

Приведите синтаксис любого класса, в котором отображалось бы свойство наследования.

Приведите синтаксис любого класса, в котором отображалось бы свойство полиморфизма.

Собеседование 4. Основы объектно-ориентированного анализа и проектирования.

- Что такое объектная модель?
- Как строится первоначальный список классов программы?
- Каким образом можно описать предполагаемые функции взаимодействия между объектами в разрабатываемой структуре?
- Укажите существующие отношения между выделенными объектами.
- Дайте определение и приведите примеры для отношений ассоциация, зависимость, обобщение и реализация.
- Какими отношениями могут быть связаны между собой объекты «Фигура» и «Конус»?

Практико-ориентированные задания для проверки умений и навыков по теме 4

Приведите примеры отношений зависимости, обобщения, ассоциации и реализации.
Изобразите графически отношение «зависимость» на примере объектов «пользователь» и «администратор».
Изобразите графически отношение «обобщение» на примере объектов «врач», «хирург» и «нейрохирург».
Изобразите графически отношение «реализация» на примере объектов «кассир» и «чек».
Изобразите графически отношение «ассоциация» на примере объектов «целый тип» и «массив данных целого типа».

Собеседование 5. Многократное использование проектов. Шаблоны проектирования

- В чем важность повторного использования?
- Какие элементы проекта могут быть повторно использованы?
- Что такое шаблон проектирования?
- Какие элементы входят в шаблон проектирования?
- Как правильно выбрать шаблон проектирования?
- Гарантирует ли использование шаблонов высокую долю повторного использования?

Практико-ориентированные задания для проверки умений и навыков по теме 5

Приведите любой практический пример применения шаблонов проектирования. Укажите элементы шаблонов проектирования.

Продемонстрируйте возможность повторного использования элементов объектно-ориентированных программных проектов при разработке классов на конкретном практическом примере.

Собеседование 6. Модульное тестирование объектно-ориентированного программного обеспечения.

- В чем состоят проблемы модульного тестирования?
- Что такое тестирование «от кода»?
- Что такое тестирование «от спецификаций»?
- В чем различия методов тестирования «прозрачного ящика» и «черного ящика»?
- Какие формы тестирования Вам известны?
- Что такое модульное тестирование?
- На каких фазах проекта следует тестировать продукт?
- Каковы пути обеспечения качества программного продукта в ходе написания кода?

Практико-ориентированные задания для проверки умений и навыков по теме 6

Осуществите модульное тестирование объектно-ориентированного программного обеспечения на примере задания из лабораторной работы. Измените начальные данные. Произведите тестирование с изначально неправильными данными. Объясните полученные результаты. Продемонстрируйте навыки обнаружения пользовательских ошибок и обработки их в программном коде.

Собеседование 7. Основы документирования объектно-ориентированного программного обеспечения

- На каких этапах жизненного цикла целесообразно документирование?
- Какие элементы проектной документации Вам известны?
- Какие программные средства помогают автоматизировать документирование?
- Какие виды документации программных проектов Вам известны?
- Что такое руководство пользователя? Каковы его необходимые элементы?
- Что такое руководство администратора? Каковы его необходимые элементы?
- Какие средства применяются для автоматизации документирования?

Практико-ориентированные задания для проверки умений и навыков по теме 7

Приведите краткое обобщение всех этапов жизненного цикла программного обеспечения. Продемонстрируйте навыки работы с программными средствами для документирования ПО. Приведите пример руководства пользователя.

Темы контрольных работ

- 1) Принципы объектно-ориентированного подхода. Примеры реализации классов.
- 2) Демонстрация и анализ принципов объектно-ориентированного программирования при разработке классов.
- 3) Понятие абстракции в математике и программировании. Примеры использования абстракции
- 4) Демонстрация и анализ концепции инкапсуляции при разработке классов
- 5) Наследование. Особенности реализации концепции наследования в языках программирования.
- 6) Полиморфизм. Демонстрация и анализ концепции полиморфизма при разработке классов.
- 7) Объектно-ориентированное проектирование. Описание структуры, отношений и взаимодействия объектов.
- 8) Дружественные функции классов. Примеры применения.
- 9) Многократное использование проектов. Примеры применения.
- 10) Анализ повторного использования элементов объектно-ориентированных программных проектов при разработке классов.
- 11) Модульное, сборочное и приемочное тестирование в объектно-ориентированном программировании.
- 12) Анализ процессов документирования при разработке классов.

Промежуточная аттестация студентов по итогам освоения дисциплины осуществляется в виде экзамена в первом семестре и дифференцированного зачета во втором семестре. Набранные во время аттестации баллы заключены в интервале от 0 до 40.

Результатом экзамена и зачета является сумма баллов текущего контроля и баллов, набранных на зачете/экзамене. Количество баллов по итогам зачет/экзамена варьируется от 60 до 100.

7. Учебно-методическое и информационное обеспечение дисциплины Б1.2.6 «Объектно-ориентированный анализ и программирование»

а) основная литература:

1. Объектно-ориентированное программирование на C++: Учебное пособие / А.А. Андрианова, Л.Н. Исмагилов, Т.М. Мухтарова. - Казань: Казанский (Приволжский) федеральный университет, 2010. – 230 с. – Режим доступа: <http://window.edu.ru/resource/950/79950>
2. Дейл, Н. Программирование на C++ [Электронный ресурс]: самоучитель / Н. Дейл, Ч. Уимз, М. Хедингтон. – Электрон. дан. – Москва: ДМК Пресс, 2007. – 672 с. – Режим доступа: <https://e.lanbook.com/book/1219>
3. Страуструп, Б. Дизайн и эволюция C++ [Электронный ресурс] – Электрон. дан. – Москва: ДМК Пресс, 2007. – 448 с. – Режим доступа: <https://e.lanbook.com/book/1222>

б) дополнительная литература:

1. Липпман, С. Язык программирования C++. Полное руководство [Электронный ресурс]: рук. / С. Липпман, Ж. Лажойе. – Электрон. дан. – Москва : ДМК Пресс, 2006. – 1105 с. – Режим доступа: <https://e.lanbook.com/book/1216>
2. Бабушкина И.А., Окулов С.М. Практикум по объектно-ориентированному

программированию. 2-е изд. – М.: БИНОМ. Лаборатория знаний, 2009. – 366 с. – Режим доступа: <http://window.edu.ru/resource/292/65292>

3. Чернышов М.К. Основы языка программирования С++ с применением объектно-ориентированного программирования: Учебно-методическое пособие. – Воронеж: ИПЦ ВГУ, 2007. - 68 с. – Режим доступа: <http://window.edu.ru/resource/356/59356>

в) программное обеспечение и Интернет-ресурсы


1. Операционная система Windows версии XP и выше
2. Microsoft Visual Studio 2010
3. MS Office 2003 и выше
4. Файловая оболочка типа Unreal Commander (свободно распространяемое ПО)
5. ЭБС «Лань»: e.lanbook.com
6. Федеральный образовательный портал «Единое окно доступа к образовательным ресурсам»: window.edu.ru
7. Интернет-университет информационных технологий www.intuit.ru

8. Материально-техническое обеспечение учебной дисциплины

1. Компьютерные и лекционные учебные классы;
2. Пакеты прикладных программ;
3. Слайды, учебные видеофильмы, плакаты.

Рабочая программа дисциплины «Объектно-ориентированный анализ и программирование» составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 38.03.05 «Бизнес-информатика».

Программу составили:

1. Абрамова Т.А., ст. преподаватель 
(Ф.И.О., должность, подпись)
2. _____
(Ф.И.О., должность, подпись)

Настоящая программа не может быть воспроизведена ни в какой форме без предварительного письменного разрешения кафедры-разработчика программы.

Программа одобрена на заседании кафедры Экономическая кибернетика


Протокол № 1 от « 31 » августа 2016 года

Зав. кафедрой ЭК  Федотов Н.Г.
(подпись, Ф.И.О.)

Программа согласована с заведующим выпускающей кафедрой
«Экономическая кибернетика» _____ Федотов Н.Г.
(название кафедры) (подпись, Ф.И.О., дата)

Программа одобрена методической комиссией факультета экономики и управления.

Протокол № 1 от « 12 » сентября 2016 года

Председатель методической комиссии
факультета экономики и управления  Еремина Е.В.
(подпись) (Ф.И.О.)

Сведения о переутверждении программы на очередной учебный год и регистрации изменений

Учебный год	Решение кафедры (№ протокола, дата, подписав. кафедрой)	Внесенные изменения	Номера листов (страниц)		
			замененных	новых	аннулированных
2017/18	№1 от 1 сентября 2017г. переутвердить без изменений <i>А.В. Герасим</i>				