

**Приложение 2 к РПД Объектно-ориентированный анализ и
программирование
09.03.02 Информационные системы и технологии
Направленность (профиль) – Информационные системы и технологии
Форма обучения – очная
Год набора - 2015**

**ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ
АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

1. Общие сведения

1.	Кафедра	Информатики и вычислительной техники
2.	Направление подготовки	09.03.02 Информационные системы и технологии
3.	Направленность (профиль)	Информационные системы и технологии
4.	Дисциплина (модуль)	Объектно-ориентированный анализ и программирование
5.	Форма обучения	очная
6.	Год набора	2015

2. Перечень компетенций

- | |
|---|
| <ul style="list-style-type: none">– владеть широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий (ОПК-1);– способность использовать математические методы обработки, анализа и синтеза результатов профессиональных исследований (ПК-25). |
|---|

3. Критерии и показатели оценивания компетенций на различных этапах их формирования

Этап формирования компетенции (разделы, темы дисциплины)	Формируемая компетенция	Критерии и показатели оценивания компетенций			Формы контроля сформированности компетенций
		Знать:	Уметь:	Владеть:	
1. Объектная модель. Принципы объектно-ориентированного подхода. Объектно-ориентированный анализ.	ОПК-1 ПК-25	Алфавит, синтаксис, семантика языка программирования. Методы описания синтаксических конструкций языков программирования. Эволюция языков программирования. Классификация языков программирования. Парадигмы программирования.	ставить задачу и разрабатывать алгоритм ее решения, использовать прикладные системы программирования	методами и инструментальными средствами разработки программ	Тест
2. Классификация и объектно-ориентированное проектирование.	ОПК-1 ПК-25	методы определения классов	читать UML нотацию диаграмм классов	современными методами выделения классов из предметной области	Тест
3. Качество классов и объектов.	ОПК-1 ПК-25	основные критерии качества абстракций: зацепление, связность, достаточность, полнота, примитивность	Применять метрики объектно-ориентированных программных систем для определения качества классов	терминологической базой	Тест
4. Объектно-ориентированные системы программирования.	ОПК-1 ПК-25	Понятия системы программирования. Техника разработки программ. Классификация ошибок в программе. Отладка программы. Основные понятия отладчика.	работать с современными системами программирования, включая объектно-ориентированные	языками процедурного и объектно-ориентированного программирования	Тест, подготовка презентации
5. Введение в язык программирования C++	ОПК-1 ПК-25	Пример простейшей программы. Операторы. Выражения. Лексемы. Типы, переменные, константы. Заголовочные файлы. Общий вид C++-программы. Понятие	уметь использовать языки системы программирования для решения профессиональных задач, работать с программными средствами общего назначения	навыками разработки и отладки программ	Лабораторная работа, групповая дискуссия

Этап формирования компетенции (разделы, темы дисциплины)	Формируемая компетенция	Критерии и показатели оценивания компетенций			Формы контроля сформированности компетенций
		Знать:	Уметь:	Владеть:	
		подпрограммы. Разработка собственных модулей Классификацию файлов. Основные операции с файлами. Классификацию структур данных.			
6. Объектно-ориентированное программирование на C++.	ОПК-1 ПК-25	Основные понятия и принципы ООП. Классы. Инкапсуляция: скрытые поля и методы. Наследование: перекрытие методов. Полиморфизм: виртуальные методы. Конструкторы и деструкторы. Динамические объекты.	решать задачи по основным разделам курса	языками процедурного и объектно-ориентированного программирования	Лабораторная работа, групповая дискуссия
7. Поточковые классы в языке C++.	ОПК-1 ПК-25	понятие потока и операций работы с ним. Стандартные потоки. Форматирование данных. методы обмена с потоками. Ошибки потоков. Файловые потоки. виды потоковых устройств. Потоки и типы, определенные пользователем.	решать задачи по основным разделам курса	языками процедурного и объектно-ориентированного программирования	Лабораторная работа, групповая дискуссия
8. Обработка исключительных ситуаций на C++.	ОПК-1 ПК-25	понятие и использование исключительной ситуации. общий механизм и синтаксис обработки исключений.	решать задачи по основным разделам курса	языками процедурного и объектно-ориентированного программирования	Лабораторная работа, групповая дискуссия
9. Шаблоны функций и классов в языке C++.	ОПК-1 ПК-25	параметризация алгоритмов и типов данных. Применение шаблонов к типам, определяемым пользователем.	решать задачи по основным разделам курса	языками процедурного и объектно-ориентированного программирования	Тест

Этап формирования компетенции (разделы, темы дисциплины)	Формируемая компетенция	Критерии и показатели оценивания компетенций			Формы контроля сформированности компетенций
		Знать:	Уметь:	Владеть:	
10. Платформа .NET Framework.	ОПК-1 ПК-25	структуру платформы .NET; принципы межъязыкового взаимодействия в .NET	создавать объектно-ориентированные приложения в среде .NET	навыком создания приложений в среде .NET	Лабораторная работа, групповая дискуссия
11. Язык программирования C#.	ОПК-1 ПК-25	основные функциональные возможности языка C#; понятие и роль языка CIL; типы данных в C#.	создавать приложения .NET с использованием Visual Studio; выполнять обработку исключений в .NET:	методикой создания динамически подключаемых библиотек в среде .NET; принципами обработки исключительных ситуаций в среде .NET	Лабораторная работа, групповая дискуссия
12. Делегаты и события в C#. Многопоточные приложения в C#.	ОПК-1 ПК-25	принципы работы с делегатами и событиями в языке C#; принципы создания многопоточных приложений	использовать механизмы синхронизации при разработке многопоточных приложений	навыками разработки многопоточных приложений	Лабораторная работа, групповая дискуссия
13. Разработка графических пользовательских интерфейсов.	ОПК-1 ПК-25	типы пользовательских интерфейсов и этапы их разработки процедурные и объектно-ориентированные пользовательские интерфейсы;	разрабатывать различные типы диалогов пользовательского интерфейса	средствами разработки графических интерфейсов пользователя в среде программирования Visual Studio	Лабораторная работа, групповая дискуссия
14. Тестирование объектно-ориентированных систем.	ОПК-1 ПК-25	основные принципы организации процесса тестирования ПО	выполнять тестирование модулей и тестирование интеграции в объектно-ориентированной разработке приложений	навыками работы с современными средствами тестирования и отладки	Тест
15. Паттерны проектирования.	ОПК-1 ПК-25	принципы использования паттернов проектирования в программных проектах	использовать паттерны в проектных решениях	навыками разработки диаграмм классов в современных case-средствах	Тест
16. Аспектно-ориентированное программирование.	ОПК-1 ПК-25	основные принципы аспектно-ориентированное программирование (АОП); преимущества и недостатки использования АОП	выделять сквозную функциональности в проектируемой системе.	технологией использования АОП в языке C#	Тест, подготовка презентации

4. Критерии и шкалы оценивания

4.1. Тесты

Процент правильных ответов	До 60	61-80	81-100
Количество баллов за решенный тест	0	1	3

4.2. Выполнение лабораторной работы (в виде программы)

10 баллов выставляется, если обучающийся выполнил все лабораторные работы, правильно изложил все варианты решения, аргументировав их, с обязательной ссылкой на соответствующие нормативы (если по содержанию это необходимо) в поставленные сроки.

4 балла выставляется, если обучающийся решил не менее 85% рекомендованных задач, правильно изложил все варианты решения, аргументировав их, с обязательной ссылкой на соответствующие нормативы (если по содержанию это необходимо).

2 балла выставляется, если обучающийся решил не менее 65% рекомендованных задач, правильно изложил все варианты их решения, аргументировав их, с обязательной ссылкой на соответствующие нормативы (если по содержанию это необходимо).

0 баллов - если обучающийся выполнил менее 50% задания, и/или неверно указал варианты решения.

4.3. Групповая дискуссия (устные обсуждения проблемы или ситуации)

Критерии оценивания	Баллы
– обучающийся ориентируется в проблеме обсуждения, грамотно высказывает и обосновывает свои суждения, владеет профессиональной терминологией, осознанно применяет теоретические знания, материал излагает логично, грамотно, без ошибок; – при ответе обучающийся демонстрирует связь теории с практикой.	2
– обучающийся грамотно излагает материал; ориентируется в проблеме обсуждения, владеет профессиональной терминологией, осознанно применяет теоретические знания, но содержание и форма ответа имеют отдельные неточности; – ответ правильный, полный, с незначительными неточностями или недостаточно полный.	1
– обучающийся излагает материал неполно, непоследовательно, допускает неточности в определении понятий, не может доказательно обосновать свои суждения; – обнаруживается недостаточно глубокое понимание изученного материала.	0

4.4. Презентация

Критерии оценки презентации	Максимальное количество баллов
Содержание (конкретно сформулирована цель работы, понятны задачи и ход работы, информация изложена полно и четко, сделаны аргументированные выводы)	4
Оформление презентации (единый стиль оформления; текст легко читается; фон сочетается с текстом и графикой; все параметры шрифта хорошо подобраны; размер шрифта оптимальный и одинаковый на всех слайдах; ключевые слова в	4

Критерии оценки презентации	Максимальное количество баллов
тексте выделены; иллюстрации усиливают эффект восприятия текстовой части информации)	
Эффект презентации (общее впечатление от просмотра презентации)	2
Максимальное количество баллов	10

4.5. Выполнение задания на составление глоссария

	Критерии оценки	Количество баллов
1	аккуратность и грамотность изложения, работа соответствует по оформлению всем требованиям	2
2	полнота исследования темы, содержание глоссария соответствует заданной теме	3
	ИТОГО:	5 баллов

5. Типовые контрольные задания и методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

1.1. Типовое тестовое задание

1. Какая компонента платформы .NET отвечает за автоматическое обнаружение, загрузку и управление типами .NET?

A: Common Language Runtime

B: Common Type System

C: Common Language Specification

2. Укажите истинные утверждения.

A: В языке программирования C# управление памятью осуществляется автоматически посредством сборки мусора.

B: В языке программирования C# не поддерживается технология аспектно-ориентированного программирования, через атрибуты.

C: В языке программирования C# предлагаются формальные синтаксические конструкции для делегатов.

D: Код, ориентируемый на выполнение в исполняющей среде .NET, называется неуправляемым кодом.

3. Возможность определения единственного имени для процедуры или функции, которые применяются ко всем объектам иерархии наследования, является:

A: Инкапсуляцией.

B: Полиморфизмом.

C: Наследованием.

4. Комбинирование данных с процедурами и функциями, манипулирующими этими данными, это следствие:

A: Наследования.

B: Полиморфизма.

C: Инкапсуляции.

5: Какому принципу объектно-ориентированного программирования удовлетворяет обоняние собаки если: "Собака чует кошку - она лает, собака чует пищу - она бежит к миске"?

- A: Наследование.
- B: Полиморфизм.
- C: Инкапсуляция.
- D: Абстракция.

6: Почему в некоторых языках программирования отказываются от поддержки множественного наследования (имеется в виду наследование реализации)?

- A: Множественное наследование невозможно реализовать с помощью таблицы виртуальных функций, поэтому требуются другие намного более сложные алгоритмы.
- B: Поддержка множественного наследования ведет к большим потерям производительности, так как для каждого класса необходимо держать сильно-ветвящуюся иерархию его предков.
- C: Из-за неоднозначности выбора поведения, в случае если суперклассы некоторого класса содержат методы с одинаковыми сигнатурами.
- D: Множественное наследование практически никогда не используется, в отличие от обычного наследования от одного класса.

7: Какой принцип объектно-ориентированного программирования необходимо использовать, чтобы заменить конструкции if-then-else в данном фрагменте кода:

```
if (animal.IsCat()) { /* код */ }
else if (animal.IsDog()) { /* код */ }
else if (animal.IsKoala()) { /* код */ }
. . .
else if (animal.isMouse()) { /* код */ }
```

- A: Наследование.
- B: Полиморфизм.
- C: Инкапсуляции.
- D: Абстракция.

8: Выберите наиболее точное определение наследованию:

- A: это механизм, который объединяет данные и методы, манипулирующие этими данными, и защищает и то и другое от внешнего вмешательства или неправильного использования.
- B: это принцип ООП, согласно которому каждый объект может использоваться более чем в одной программе.
- C: это механизм, позволяющий создавать классы объектов на основе других классов, расширяя и частично изменяя их функциональность и набор атрибутов.
- D: это механизм, который позволяет описывать новые классы на основании других классов

9: Какой принцип ООП нарушает следующий фрагмент кода:

```

class Counter {
    public int count;
    public void increment() { count++; }
    public int get() { return count; }
}
public class Main {
    public static void main(String[] args) {
        Counter counter = new Counter();
        counter.count = 5;
    }
}

```

- A: Наследование.
- B: Полиморфизм.
- C: Инкапсуляции.
- D: Абстракция.

10: Какая разница между объектом и классом?

- A: Класс - это исходный код, а объект - скомпилированный и выполняемый код.
- B: Класс описывает категорию, к которой могут либо принадлежать, либо не принадлежать объекты данного класса.
- C: Класс может иметь много экземпляров, а объект - один или ни одного.
- D: Объект - это экземпляр класса.

Ключ: 1-A, 2-A, 3-B, 4-C, 5-B, 6-C, 7-B, 8-C, 9-C, 10-C

1.2. Пример лабораторной работы

Тема лабораторная работы: Классы и объекты в C++.

Ход выполнения работы:

1. Определить пользовательский класс в соответствии с вариантом.
2. Определить в классе следующие конструкторы: без параметров, с параметрами, копирования.
3. Определить в классе деструктор.
4. Определить в классе компоненты-функции для просмотра и установки полей данных.
5. Определить указатель на экземпляр класса.
6. Написать демонстрационную программу, в которой создаются и разрушаются объекты пользовательского класса и каждый вызов конструктора и деструктора сопровождается выдачей соответствующего сообщения (какой объект какой конструктор или деструктор вызвал). Вариант задания выдается преподавателем.
7. Показать в программе использование указателя на объект.
8. Лабораторная должна состоять из трёх файлов:
 - заголовочный h-файл с определением класса,
 - сpp-файл с реализацией класса,
 - сpp-файл демонстрационной программы.

1.3. Вопросы к экзамену:

1. Объектно-ориентированный анализ и проектирование как эволюция в развитии методов проектирования информационных систем.
2. Принципы объектно-ориентированного подхода: абстрагирование, инкапсуляция, модульность, наследование, типизация, сохраняемость, параллелизм. Классы и объекты.
3. Состояние и поведение объекта. Взаимодействие объектов.
4. Методы, классификация методов: модификаторы, селекторы, итераторы,

конструкторы, деструкторы.

5. Идентичность объекта.
6. Идентификаторы доступа к полям объекта.
7. Классификация и объектно-ориентированное проектирование.
8. Методы определения классов: классические методы.
9. Методы определения классов: анализ поведения.
10. Методы определения классов: анализ предметной области.
11. Методы определения классов: анализ вариантов.
12. Методы определения классов: CRC карточки.
13. Методы определения классов: неформальное описание.
14. Методы определения классов: структурный анализ.
15. Диаграмма классов в нотации UML.
16. Критерии качества абстракций: зацепление, связность.
17. Критерии качества абстракций: достаточность, полнота, примитивность.
18. Выбор операция класса.
19. Метрики объектно-ориентированных программных систем: недостаток связности в методах.
20. Метрики объектно-ориентированных программных систем: процент публичных и защищенных.
21. Метрики объектно-ориентированных программных систем: публичный доступ к компонентным данным.
22. Метрики объектно-ориентированных программных систем: количество корневых классов.
23. Метрики объектно-ориентированных программных систем: высота дерева наследования.
24. Метрики объектно-ориентированных программных систем: количество детей.
25. Метрики объектно-ориентированных программных систем: фактор полиморфизма.
26. Понятия системы программирования. Техника разработки программ. Классификация ошибок в программе.
27. Отладка программы. Основные понятия отладчика.
28. Обзор современных объектно-ориентированных систем программирования.
29. Основные понятия языка программирования C++: лексемы, константы, типы данных, выражения.
30. Основные понятия языка программирования C++: инструкции (операторы), блочная структура программы на C++.
31. Распределение памяти.
32. Процедуры и функции. Описание подпрограмм.
33. Способы передачи параметров в подпрограмму и *возвращение результата*.
Процедурные типы.
34. Заголовочные файлы языка C++. Основные операции с файлами в языке C++.
35. Реализация структур данных на языке C++: стек.
36. Реализация структур данных на языке C++: дек.
37. Реализация структур данных на языке C++: очередь.
38. Основные понятия и принципы ООП. Классы.
39. Инкапсуляция: скрытые поля и методы.
40. Наследование: перекрытие методов.
41. Полиморфизм: виртуальные методы.
42. Конструкторы и деструкторы.
43. Динамические объекты.
44. Понятие потока и операций работы с ним. Стандартные потоки.
45. Форматирование данных, методы обмена с потоками.
46. Ошибки потоков.

47. Файловые потоки. виды потоковых устройств.
48. Потоки и типы, определенные пользователем.
49. Понятие и использование исключительной ситуации.
50. Общий механизм и синтаксис обработки исключений в C++.
51. Параметризация алгоритмов и типов данных.
52. Применение шаблонов к типам, определяемым пользователем в C++.

ТЕХНОЛОГИЧЕСКАЯ КАРТА ДИСЦИПЛИНЫ

ОСНОВНАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА

09.03.02 Информационные системы и технологии

Направленность (профиль) «Информационные системы и технологии»

(код, направление, профиль)

ТЕХНОЛОГИЧЕСКАЯ КАРТА

Шифр дисциплины по РУП		Б1.В.ДВ.6.2					
Дисциплина		Объектно-ориентированный анализ и программирование					
Курс	2	семестр	3				
Кафедра		Информатики и вычислительной техники					
Ф.И.О. преподавателя, звание, должность		Тоичкин Николай Александрович, канд. техн. наук, доцент кафедры информатики и вычислительной техники					
Общ. трудоемкость ^{час/ЗЕТ}		108/3	Кол-во семестров	1	Форма контроля	Экзамен	
ЛК _{общ./тек. сем.}	66/32	ПР/СМ _{общ./тек. сем.}	-/-	ЛБ _{общ./тек. сем.}	66/32	СРС _{общ./тек. сем.}	12/8

Компетенции обучающегося, формируемые в результате освоения дисциплины:

- владеть широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий (ОПК-1);
- способность использовать математические методы обработки, анализа и синтеза результатов профессиональных исследований (ПК-25).

Код формируемой компетенции	Содержание задания	Количество мероприятий	Максимальное количество баллов	Срок предоставления
Вводный блок				
Не предусмотрен				
Основной блок				
ОПК-1 ПК-25	Решение тестов	4	12	В течение семестра по расписанию занятий
ОПК-1 ПК-25	Лабораторные работы	4	40	В течение семестра по расписанию занятий
ОПК-1 ПК-25	Групповые дискуссии	4	8	В течение семестра по расписанию занятий
ОПК-1 ПК-25	Подготовка презентации	1	10	В течение семестра по расписанию занятий
Всего:			60	
ОПК-1 ПК-25	Экзамен	Вопрос 1	20	В сроки сессии
		Вопрос 2	20	В сроки сессии
Всего:			40	
Итого:			100	
Дополнительный блок				
ОПК-1	Выполнение дополнительной лабораторной работы		10	по согласованию с преподавателем
ПК-25	Подготовка глоссария		5	
Всего:			15	

Шкала оценивания в рамках балльно-рейтинговой системы МАГУ: «2» - 60 баллов и менее, «3» - 61-80 баллов, «4» - 81-90 баллов, «5» - 91-100 баллов.

ТЕХНОЛОГИЧЕСКАЯ КАРТА

Шифр дисциплины по РУП		Б1.В.ДВ.6.2			
Дисциплина		Объектно-ориентированный анализ и программирование			
Курс	2	семестр	4		
Кафедра	Информатики и вычислительной техники				
Ф.И.О. преподавателя, звание, должность		Тоичкин Николай Александрович, канд. техн. наук, доцент кафедры информатики и вычислительной техники			
Общ. трудоемкость _{час/ЗЕТ}		108/3	Кол-во семестров	1	Форма контроля
ЛК _{общ./тек. сем.}		66/34	ПР/СМ _{общ./тек. сем.}	-/-	ЛБ _{общ./тек. сем.}
				66/34	СРС _{общ./тек. сем.}
					12/4

Компетенции обучающегося, формируемые в результате освоения дисциплины:

- владеть широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий (ОПК-1);
- способность использовать математические методы обработки, анализа и синтеза результатов профессиональных исследований (ПК-25).

Код формируемой компетенции	Содержание задания	Количество мероприятий	Максимальное количество баллов	Срок предоставления
<i>Вводный блок</i>				
Не предусмотрен				
<i>Основной блок</i>				
ОПК-1 ПК-25	Решение тестов	4	12	В течение семестра
ОПК-1 ПК-25	Лабораторные работы	4	40	В течение семестра по расписанию занятий
ОПК-1 ПК-25	Групповые дискуссии	4	8	В течение семестра по расписанию занятий
ОПК-1 ПК-25	Подготовка презентации	1	10	В течение семестра по расписанию занятий
Всего:			60	
ОПК-1 ПК-25	Экзамен	Вопрос 1	20	В сроки сессии
		Вопрос 2	20	В сроки сессии
Всего:			40	
Итого:			100	
<i>Дополнительный блок</i>				
ОПК-1	Выполнение дополнительной лабораторной работы		10	по согласованию с преподавателем
ПК-25	Подготовка глоссария		5	
Всего:			15	

Шкала оценивания в рамках балльно-рейтинговой системы МАГУ: «2» - 60 баллов и менее, «3» - 61-80 баллов, «4» - 81-90 баллов, «5» - 91-100 баллов.