

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное учреждение  
высшего профессионального образования  
"Казанский (Приволжский) федеральный университет"  
Институт вычислительной математики и информационных технологий



**Программа дисциплины**

Объектно-ориентированный анализ и программирование Б1.Б.18

Направление подготовки: 38.03.05 - Бизнес-информатика

Профиль подготовки:

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

**Автор(ы):**

Андрианова А.А. , Мухтарова Т.М.

**Рецензент(ы):**

Тагиров Р.Р. , Миссаров М.Д.

**СОГЛАСОВАНО:**

Заведующий(ая) кафедрой: Латыпов Р. Х.

Протокол заседания кафедры No \_\_\_\_ от " \_\_\_\_ " \_\_\_\_\_ 201\_\_г

Учебно-методическая комиссия Института вычислительной математики и информационных технологий:

Протокол заседания УМК No \_\_\_\_ от " \_\_\_\_ " \_\_\_\_\_ 201\_\_г

Регистрационный No 92716

Казань

2016

## **Содержание**

1. Цели освоения дисциплины
2. Место дисциплины в структуре основной образовательной программы
3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля
4. Структура и содержание дисциплины/ модуля
5. Образовательные технологии, включая интерактивные формы обучения
6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов
7. Литература
8. Интернет-ресурсы
9. Материально-техническое обеспечение дисциплины/модуля согласно утвержденному учебному плану

Программу дисциплины разработал(а)(и) доцент, к.н. Андрианова А.А. кафедра системного анализа и информационных технологий отделение фундаментальной информатики и информационных технологий , Anastasiya.Andrianova@kpfu.ru ; старший преподаватель, б/с Мухтарова Т.М. кафедра анализа данных и исследования операций отделение фундаментальной информатики и информационных технологий , Tatyana.Moukhtarova@kpfu.ru

## 1. Цели освоения дисциплины

Данная дисциплина предполагает изучение одной из современных технологий программирования - объектно-ориентированной технологии, на основе которой строится большинство современных информационных систем. Данная технология изучается на примере языка программирования C#.

Практические занятия по курсу производятся с помощью среды разработки Visual Studio 2005 (2008, 2010).

## 2. Место дисциплины в структуре основной образовательной программы высшего профессионального образования

Данная учебная дисциплина включена в раздел " Б1.Б.18 Дисциплины (модули)" основной образовательной программы 38.03.05 Бизнес-информатика и относится к базовой (общепрофессиональной) части. Осваивается на 2 курсе, 3 семестр.

Дисциплина "Объектно-ориентированный анализ и программирование" изучается на 2 курсе в 3 семестре обучения бакалавров. Предварительно студенты должны получить базовые знания и навыки по программированию при изучении дисциплины "Программирование". В дальнейшем знания по курсу "Объектно-ориентированный анализ и программирование" потребуются при изучении курсов "Интернет-технологии", "Вычислительные системы, сети и телекоммуникации", "Проектирование информационных систем" и других учебных дисциплин, которые предусмотрены учебным планом по направлению "Бизнес-информатика". Навыки, полученные при изучении этого предмета, будут использованы студентами в рамках учебной и производственной практики, при написании курсовых и дипломных работ.

## 3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля

В результате освоения дисциплины формируются следующие компетенции:

Шифр компетенции	Расшифровка приобретаемой компетенции
ОК-12 (общекультурные компетенции)	осознает сущность и значение информации в развитии современного общества; владеет основными методами, способами и средствами получения, хранения, переработки информации;
ПК-18 (профессиональные компетенции)	разрабатывать контент и ИТ-сервисы предприятия и Интернет-ресурсов;
ПК-21 (профессиональные компетенции)	готовить научно-технические отчеты, презентации, научные публикации по результатам выполненных исследований.

В результате освоения дисциплины студент:

1. должен знать:

- основные принципы объектно-ориентированного программирования.
- основные синтаксические правила языка программирования C#, которые используются для реализации принципов объектно-ориентированного программирования;

- основные библиотеки классов языка программирования C# в целях использования библиотечных классов для создания объектно-ориентированных приложений.

2. должен уметь:

- создавать собственные классы на языке программирования C#, а также использовать классы из библиотек этого языка;
- создавать сложные приложения с использованием расширенного пользовательского интерфейса и привлечением внешних источников данных.

3. должен владеть:

- навыками разработки консольных приложений в стиле объектно-ориентированного программирования на языке программирования C#;
  - навыками разработки Windows-приложений в стиле объектно-ориентированного программирования на языке программирования C#;
  - навыками работы приложений с внешними источниками данных (текстовыми файлами, xml-файлами, базами данных).
- применять полученные знания и навыки в процессе дальнейшего обучения, при написании курсовых и дипломной работы, а также в своей дальнейшей профессиональной деятельности

#### **4. Структура и содержание дисциплины/ модуля**

Общая трудоемкость дисциплины составляет 4 зачетных(ые) единиц(ы) 144 часа(ов).

Форма промежуточного контроля дисциплины экзамен в 3 семестре.

Суммарно по дисциплине можно получить 100 баллов, из них текущая работа оценивается в 50 баллов, итоговая форма контроля - в 50 баллов. Минимальное количество для допуска к зачету 28 баллов.

86 баллов и более - "отлично" (отл.);

71-85 баллов - "хорошо" (хор.);

55-70 баллов - "удовлетворительно" (удов.);

54 балла и менее - "неудовлетворительно" (неуд.).

#### **4.1 Структура и содержание аудиторной работы по дисциплине/ модулю**

##### **Тематический план дисциплины/модуля**

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практические занятия	Лабораторные работы	
1.	Тема 1. Понятие класса. Инкапсуляция. Члены класса - переменные (данные-члены) и методы (функции-члены) класса. Управление доступом к членам класса. Операции доступа к членам класса.	3	1	4	0	2	домашнее задание
2.	Тема 2. Конструкторы и деструкторы. Назначение конструкторов и деструкторов. Отличия конструкторов и деструкторов от обычных функций-членов класса. Конструкторы по умолчанию. Необходимость перегрузки конструкторов.	3	2	2	0	1	домашнее задание
3.	Тема 3. Свойства классов. Внедрение объектов. Особенности вызова конструкторов внедренных объектов. Вложенные классы.	3	3	2	0	1	домашнее задание
4.	Тема 4. Перегрузка операций в классе. Общие правила переопределения операций. Перегрузка бинарных операций. Перегрузка унарных операций. Особенности перегрузки префиксной и постфиксной форм операции инкремента.	3	4	4	0	3	домашнее задание

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практические занятия	Лабораторные работы	
5.	Тема 5. Наследование. Понятие наследования. Базовый и производный классы. Порядок вызова конструкторов и деструкторов при наследовании. Управление доступом при наследовании. Спецификатор доступа protected.	3	5-6	4	0	3	домашнее задание
6.	Тема 6. Полиморфизм и виртуальные функции. Особенности использования указателей на объект базового класса. Виртуальные функции как основной способ реализации полиморфизма. Основные примеры использования виртуальных функций. Абстрактные методы и классы, интерфейсы.	3	7-9	4	0	3	домашнее задание
8.	Тема 8. Обработка исключительных ситуаций. Возможности программной обработки исключительных ситуаций. Операторы try, catch, throw. Создание иерархии исключений.	3	10	2	0	2	домашнее задание
9.	Тема 9. Объектно-ориентированные особенности языка C# Свойства. Индексаторы. Итераторы. Статические члены класса.	3	11-12	2	0	2	контрольная работа домашнее задание

N	Раздел Дисциплины/ Модуля	Семестр	Неделя семестра	Виды и часы аудиторной работы, их трудоемкость (в часах)			Текущие формы контроля
				Лекции	Практические занятия	Лабораторные работы	
10.	Тема 10. Делегаты и события Определение простого и комбинированного делегата. Применение делегатов. Использование делегатов для реализации событий. Примеры использования событий.	3	13-14	3	0	1	домашнее задание
11.	Тема 11. Работа с коллекциями Классы из пространства имен Collections. Коллекции-контейнеры - динамические массивы, списки, словари. Использование нумераторов. Сравнение элементов коллекций. Коллекции-шаблоны.	3	15-16	3	0	0	домашнее задание
12.	Тема 12. Работа с файлами Потоки ввода. Потоки вывода. Текстовые и двоичные файлы.	3	17	3	0	0	домашнее задание
13.	Тема 13. Обобщенные классы (шаблоны). Понятие шаблона класса. Использование шаблона класса. Ограничения при использовании шаблонов.	3	18	3	0	0	контрольная работа домашнее задание
.	Тема . Итоговая форма контроля	3		0	0	0	экзамен
	Итого			36	0	18	

## 4.2 Содержание дисциплины

**Тема 1. Понятие класса. Инкапсуляция. Члены класса - переменные (данные-члены) и методы (функции-члены) класса. Управление доступом к членам класса. Операции доступа к членам класса.**

**лекционное занятие (4 часа(ов)):**



Понятие класса. Инкапсуляция. Члены класса - переменные (данные-члены) и методы (функции-члены) класса. Управление доступом к членам класса. Операции доступа к членам класса.

**лабораторная работа (2 часа(ов)):**

Решение задач по созданию собственных классов. Классы должны содержать переменные и методы. Обязательно должны быть функции ввода и вывода и другие методы класса. Типовые задачи: рациональная дробь, точка в пространстве, треугольник.

**Тема 2. Конструкторы и деструкторы. Назначение конструкторов и деструкторов.**

**Отличия конструкторов и деструкторов от обычных функций-членов класса.**

**Конструкторы по умолчанию. Необходимость перегрузки конструкторов.**

**лекционное занятие (2 часа(ов)):**

Конструкторы и деструкторы. Назначение конструкторов и деструкторов. Отличия конструкторов и деструкторов от обычных функций-членов класса. Конструкторы по умолчанию. Необходимость перегрузки конструкторов.

**лабораторная работа (1 часа(ов)):**

Решение задач по созданию собственных классов. В классах должны быть предусмотрены конструкторы и деструктор. Типовые задачи: вектор, матрица, множество.

**Тема 3. Свойства классов. Внедрение объектов. Особенности вызова конструкторов внедренных объектов. Вложенные классы.**

**лекционное занятие (2 часа(ов)):**

Свойства классов. Внедрение объектов. Особенности вызова конструкторов внедренных объектов. Вложенные классы.

**лабораторная работа (1 часа(ов)):**

Решение задач по созданию собственных классов с внедрением объектов. Типовые задачи: стек и очередь.

**Тема 4. Перегрузка операций в классе. Общие правила переопределения операций.**

**Перегрузка бинарных операций. Перегрузка унарных операций. Особенности перегрузки префиксной и постфиксной форм операции инкремента.**

**лекционное занятие (4 часа(ов)):**

Перегрузка операций в классе. Общие правила переопределения операций. Перегрузка бинарных операций. Перегрузка унарных операций. Особенности перегрузки префиксной и постфиксной форм операции инкремента.

**лабораторная работа (3 часа(ов)):**

Решение задач по созданию собственных классов, осуществляющих работу с математическими объектами, для которых можно переопределить операции. Типовые задачи: рациональная дробь, вектор, матрица.

**Тема 5. Наследование. Понятие наследования. Базовый и производный классы.**

**Порядок вызова конструкторов и деструкторов при наследовании. Управление доступом при наследовании. Спецификатор доступа protected.**

**лекционное занятие (4 часа(ов)):**

Наследование. Понятие наследования. Базовый и производный классы. Порядок вызова конструкторов и деструкторов при наследовании. Управление доступом при наследовании. Спецификатор доступа protected.

**лабораторная работа (3 часа(ов)):**

Решение задач по созданию собственной иерархии классов. Типовые задачи: иерархия многоугольников (треугольник-четыреугольник-пятиугольник) иерархия линейных структур данных (списки - стеки - очереди).

**Тема 6. Полиморфизм и виртуальные функции. Особенности использования указателей на объект базового класса. Виртуальные функции как основной способ реализации полиморфизма. Основные примеры использования виртуальных функций. Абстрактные методы и классы, интерфейсы.**

**лекционное занятие (4 часа(ов)):**

Полиморфизм и виртуальные функции. Особенности использования указателей на объект базового класса. Виртуальные функции как основной способ реализации полиморфизма. Основные примеры использования виртуальных функций. Абстрактные методы и классы, интерфейсы.

**лабораторная работа (3 часа(ов)):**

Решение задач по созданию собственной иерархии классов и использованию полиморфизма. Типовые задачи: иерархия многоугольников (треугольник-четыреугольник-пятиугольник) (методы печати информации и перерисовки), иерархия линейных структур данных (списки - стеки - очереди) (методы добавления и извлечения) и прочие.

**Тема 8. Обработка исключительных ситуаций. Возможности программной обработки исключительных ситуаций. Операторы try, catch, throw. Создание иерархии исключений.**

**лекционное занятие (2 часа(ов)):**

Обработка исключительных ситуаций. Возможности программной обработки исключительных ситуаций. Операторы try, catch, throw. Создание иерархии исключений.

**лабораторная работа (2 часа(ов)):**

Решение задач с собственными исключительными ситуациями. Типовые задачи: исключения, связанные с матрицами, исключения, связанные с линейными структурами данных.

**Тема 9. Объектно-ориентированные особенности языка C# Свойства. Индексаторы. Итераторы. Статические члены класса.**

**лекционное занятие (2 часа(ов)):**

Объектно-ориентированные особенности языка C# Свойства. Индексаторы. Итераторы. Статические члены класса.

**лабораторная работа (2 часа(ов)):**

Решение задач с использованием свойств, индексаторов, статических членов класса. Типовые задачи: матрица, список и прочие.

**Тема 10. Делегаты и события Определение простого и комбинированного делегата. Применение делегатов. Использование делегатов для реализации событий. Примеры использования событий.**

**лекционное занятие (3 часа(ов)):**

Делегаты и события Определение простого и комбинированного делегата. Применение делегатов. Использование делегатов для реализации событий. Примеры использования событий.

**лабораторная работа (1 часа(ов)):**

Разработка программ с использованием делегатов: нахождение корней уравнения методом деления отрезка пополам, хорд, касательных. Разработка класса с событием: класс квадратное уравнение, событие - изменение коэффициентов уравнения.

**Тема 11. Работа с коллекциями Классы из пространства имен Collections. Коллекции-контейнеры - динамические массивы, списки, словари. Использование нумераторов. Сравнение элементов коллекций. Коллекции-шаблоны.**

**лекционное занятие (3 часа(ов)):**

Работа с коллекциями Классы из пространства имен Collections. Коллекции-контейнеры - динамические массивы, списки, словари. Использование нумераторов. Сравнение элементов коллекций. Коллекции-шаблоны.

**Тема 12. Работа с файлами Потоки ввода. Потоки вывода. Текстовые и двоичные файлы.**

**лекционное занятие (3 часа(ов)):**

Работа с файлами Потоки ввода. Потоки вывода. Текстовые и двоичные файлы.

**Тема 13. Обобщенные классы (шаблоны). Понятие шаблона класса. Использование шаблона класса. Ограничения при использовании шаблонов.**

**лекционное занятие (3 часа(ов)):**

Обобщенные классы (шаблоны). Понятие шаблона класса. Использование шаблона класса. Ограничения при использовании шаблонов.

#### 4.3 Структура и содержание самостоятельной работы дисциплины (модуля)

N	Раздел Дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
1.	Тема 1. Понятие класса. Инкапсуляция. Члены класса - переменные (данные-члены) и методы (функции-члены) класса. Управление доступом к членам класса. Операции доступа к членам класса.	3	1	подготовка домашнего задания	4	домашнее задание
2.	Тема 2. Конструкторы и деструкторы. Назначение конструкторов и деструкторов. Отличия конструкторов и деструкторов от обычных функций-членов класса. Конструкторы по умолчанию. Необходимость перегрузки конструкторов.	3	2	подготовка домашнего задания	4	домашнее задание
3.	Тема 3. Свойства классов. Внедрение объектов. Особенности вызова конструкторов внедренных объектов. Вложенные классы.	3	3	подготовка домашнего задания	4	домашнее задание
4.	Тема 4. Перегрузка операций в классе. Общие правила переопределения операций. Перегрузка бинарных операций. Перегрузка унарных операций. Особенности перегрузки префиксной и постфиксной форм операции инкремента.	3	4	подготовка домашнего задания	4	домашнее задание

N	Раздел Дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
5.	Тема 5. Наследование. Понятие наследования. Базовый и производный классы. Порядок вызова конструкторов и деструкторов при наследовании. Управление доступом при наследовании. Спецификатор доступа protected.	3	5-6	подготовка домашнего задания	4	домашнее задание
6.	Тема 6. Полиморфизм и виртуальные функции. Особенности использования указателей на объект базового класса. Виртуальные функции как основной способ реализации полиморфизма. Основные примеры использования виртуальных функций. Абстрактные методы и классы, интерфейсы.	3	7-9	подготовка домашнего задания	6	домашнее задание
8.	Тема 8. Обработка исключительных ситуаций. Возможности программной обработки исключительных ситуаций. Операторы try, catch, throw. Создание иерархии исключений.	3	10	подготовка домашнего задания	4	домашнее задание
9.	Тема 9. Объектно-ориентированные особенности языка C# Свойства. Итераторы. Статические члены класса.	3	11-12	подготовка домашнего задания	2	домашнее задание
				подготовка к контрольной работе	4	контрольная работа

N	Раздел Дисциплины	Семестр	Неделя семестра	Виды самостоятельной работы студентов	Трудоемкость (в часах)	Формы контроля самостоятельной работы
10.	Тема 10. Делегаты и события Определение простого и комбинированного делегата. Применение делегатов. Использование делегатов для реализации событий. Примеры использования событий.	3	13-14	подготовка домашнего задания	6	домашнее задание
11.	Тема 11. Работа с коллекциями Классы из пространства имен Collections. Коллекции-контейнеры - динамические массивы, списки, словари. Использование нумераторов. Сравнение элементов коллекций. Коллекции-шаблоны.	3	15-16	подготовка домашнего задания	4	домашнее задание
12.	Тема 12. Работа с файлами Потоки ввода. Потоки вывода. Текстовые и двоичные файлы.	3	17	подготовка домашнего задания	4	домашнее задание
13.	Тема 13. Обобщенные классы (шаблоны). Понятие шаблона класса. Использование шаблона класса. Ограничения при использовании шаблонов.	3	18	подготовка домашнего задания	2	домашнее задание
				подготовка к контрольной работе	2	контрольная работа
	Итого				54	

## 5. Образовательные технологии, включая интерактивные формы обучения

Дисциплина представляет собой цикл лекционных и лабораторных (практических) занятий. Практические занятия посвящены выработке навыков разработки программ в объектно-ориентированном стиле на языке программирования C#. Практические занятия проходят в компьютерных классах с использованием интерактивной доски для наглядного представления алгоритмов и разработки программ на всех этапах ее создания и компиляции. Практические занятия проходят в интерактивной форме обсуждения решения различных задач или в активной форме самостоятельного решения задач студентами. Контроль за выполнением самостоятельной работы проявляется в функциональном тестировании выполненных студентами заданий на примерах, предложенных преподавателем.

## **6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов**

### **Тема 1. Понятие класса. Инкапсуляция. Члены класса - переменные (данные-члены) и методы (функции-члены) класса. Управление доступом к членам класса. Операции доступа к членам класса.**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Типовой пример задачи: разработка класса "рациональная дробь", в котором нужно предусмотреть методы ввода и вывода, методы сложения, вычитания, умножения, деления дробей, выполнения всех указанных операций тогда, когда один из операндов целое число, методы сравнения двух дробей. В качестве домашнего задания может быть разработка аналогичных по требованиям классов (прямоугольник, вектор на плоскости и т.д.)

### **Тема 2. Конструкторы и деструкторы. Назначение конструкторов и деструкторов. Отличия конструкторов и деструкторов от обычных функций-членов класса. Конструкторы по умолчанию. Необходимость перегрузки конструкторов.**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Типовой пример задачи: добавление в разработанный класс конструкторов и деструктора. В качестве примера лучше рассматривать разработку класса "вектор в n-мерном пространстве". В качестве домашнего задания может быть разработка аналогичных по требованиям классов (матрица, множество, полином и т.д.)

### **Тема 3. Свойства классов. Внедрение объектов. Особенности вызова конструкторов внедренных объектов. Вложенные классы.**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Типовой пример задачи по теме: реализация класса "Стек" с внедренным объектом, который хранится в стеке. В качестве объекта можно использовать простой класс, описывающий некоторый объект предметной области (книгу, студента и пр.). В качестве домашнего задания может быть разработка аналогичных по требованиям классов (матрица, состоящая из рациональных дробей, матрица как массив векторов, разреженная матрица).

### **Тема 4. Перегрузка операций в классе. Общие правила переопределения операций. Перегрузка бинарных операций. Перегрузка унарных операций. Особенности перегрузки префиксной и постфиксной форм операции инкремента.**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Типовой пример задачи: доработка уже разработанных классов, описывающих математические объекты, арифметическими и логическими операциями (например, для рационального числа). В качестве домашнего задания можно доработать классы математических объектов, которые были разработаны ранее, введением в их состав арифметических и других видов операций.

### **Тема 5. Наследование. Понятие наследования. Базовый и производный классы. Порядок вызова конструкторов и деструкторов при наследовании. Управление доступом при наследовании. Спецификатор доступа protected.**

домашнее задание , примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Типовая задача: иерархия многоугольников "треугольник"- "четыреугольник"- "пятиугольник". Наследование реализуется посредством наследования вершин из предшествующего вида многоугольника и операций ввода, вывода, вычисления площади и определения попадания точки в множество. В качестве домашнего задания можно разработать иерархию классов матриц (прямоугольная матрица - квадратная матрица - единичная матрица).



**Тема 6. Полиморфизм и виртуальные функции. Особенности использования указателей на объект базового класса. Виртуальные функции как основной способ реализации полиморфизма. Основные примеры использования виртуальных функций. Абстрактные методы и классы, интерфейсы.**

домашнее задание, примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Типовая задача: разработка иерархии классов-фигур: прямоугольник, круг, отрезок и т.д. Полиморфизм представляется в виде разработки виртуальных методов ввода информации и рисования фигур в пользовательском окне. В качестве домашнего задания можно разработать иерархию классов-функций (линейная-квадратичная) с виртуальными методами вычисления значения в точке, вычисления градиента и пр.

**Тема 8. Обработка исключительных ситуаций. Возможности программной обработки исключительных ситуаций. Операторы try, catch, throw. Создание иерархии исключений.**

домашнее задание, примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Типовое задание по теме: доработка классов-матриц с проверками возможности выполнения операций с матрицами и генерацией исключительных ситуаций в случае невозможности их выполнения. В качестве домашнего задания можно доработать классы-линейные структуры данных, разработанные ранее, на предмет обнаружения и обработки исключительных ситуаций.

**Тема 9. Объектно-ориентированные особенности языка C# Свойства. Индексаторы. Итераторы. Статические члены класса.**

домашнее задание, примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Типовое задание: доработка ранее разработанного класса "Матрица" с точки зрения добавления в него элементов, определяющих удобный и контролируемый доступ к данным класса (свойства и индексаторы). В качестве домашнего задания можно аналогичным образом доработать классы линейных структур данных (списки, стеки, очереди).

контрольная работа, примерные вопросы:

Пример задания на контрольную работу: Разработать класс отрезок на прямой. Определить конструкторы, операции пересечения и объединения отрезков, индексатор для получения значения каждого из концов отрезка.

**Тема 10. Делегаты и события Определение простого и комбинированного делегата. Применение делегатов. Использование делегатов для реализации событий. Примеры использования событий.**

домашнее задание, примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Решаются простые задачи: методы решения уравнений  $F(X)=0$ , методы сортировки объектов класса с различными критериями упорядочивания.

**Тема 11. Работа с коллекциями Классы из пространства имен Collections. Коллекции-контейнеры - динамические массивы, списки, словари. Использование нумераторов. Сравнение элементов коллекций. Коллекции-шаблоны.**

домашнее задание, примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Решаются задачи организации и использования собственных структур данных для хранения и реализации доступа к коллекциям объектов различной сложности (списки книг в библиотеке, телефонная книга и прочие).

**Тема 12. Работа с файлами Потоки ввода. Потоки вывода. Текстовые и двоичные файлы.**

домашнее задание, примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Решаются задачи с использованием внешних источников данных в виде текстовых и бинарных файлов. Рассматриваются операции чтения и записи в файлы. Типовые примеры - матрица, список объектов.

### **Тема 13. Обобщенные классы (шаблоны). Понятие шаблона класса. Использование шаблона класса. Ограничения при использовании шаблонов.**

домашнее задание, примерные вопросы:

Углубленное изучение литературы. Обсуждение. Решение задач по теме. Типовая задача: организация шаблона класса "Множество", которое позволяет хранить элементы различных типов данных.

контрольная работа, примерные вопросы:

### **Тема . Итоговая форма контроля**

Примерные вопросы к экзамену:

По данной дисциплине предусмотрен экзамен, который проводится в практической форме и ответами на теоретические вопросы из программы курса.

Вопросы к экзамену:

Основные понятия объектно-ориентированного анализа и программирования. Понятие объектной модели: классы, объекты, свойства. Отношения между классами. Основные принципы объектно-ориентированного программирования. Абстрагирование и инкапсуляция. Класс как способ инкапсуляции и сокрытия данных. Переменные и методы класса. Вызов методов класса. Статические элементы класса. Конструкторы, деструктор и необходимость их создания. Конструкторы с параметрами. Перегрузка конструкторов. Необходимость перегрузки. Свойства и индексы как элементы класса. Перегрузка операций как один из способов реализации полиморфизма.

Наследование как один из основных принципов объектно-ориентированного программирования. Наследование и необходимость его использования. Переопределение методов базового класса. Спецификатор доступа protected. Конструкторы и деструкторы при наследовании. Правила их вызова. Особенности преобразования типов между наследуемыми классами.

Особенность динамического полиморфизма. Особенности использования ссылок на базовые классы. Виртуальные функции как одна из форм полиморфизма. Необходимость их использования. Абстрактные функции и абстрактные классы и их использование. Создание иерархии исключений. Использование виртуальной природы функций при обработке исключений.

Делегаты и события. Понятие делегата как класса, объекты которого хранят информацию о методах. Способ вызова метода через делегат. Использование делегатов. События как способ использования делегатов.

Обобщенные функции и классы. Обобщенные функции и особенности их использования. Обобщенные классы и особенности их использования. Цели использования обобщенных функций и классов. Уточнения при создании обобщений.

Регламент проведения экзамена

по курсу "Объектно-ориентированное программирование"

Экзамен будет проходить в форме решения задачи на компьютере. Задача будет состоять в разработке одного класса или двух взаимосвязанных классов и их применении. Задачи будут подобраны так, чтобы студент мог продемонстрировать знания всех принципов объектно-ориентированного программирования.

Основные темы экзаменационных задач:

- Использование полиморфизма при перегрузке операций класса;
- Разработка иерархии классов и использование динамического полиморфизма;
- Обобщенные функции, использование делегатов и событий.

Решенная на экзамене задача оценивается в 40 баллов. Оставшиеся 10 баллов могут быть получены студентами за ответ на дополнительный теоретический вопрос из программы курса. При ответе на теоретический вопрос можно использовать практическое задание экзамена.



### Типовое задание на экзамен:

Создать класс "Измерение", который характеризуется названием измеряемого показателя, значением измерения и названием единицы измерения. Требуется организовать создание и корректную инициализацию объектов данного класса и обеспечить доступ к его элементам (7 баллов).

Создать класс "Эксперимент", который содержит массив различных измерений (температуры, влажности, силы ветра и пр.). Требуется обеспечить с помощью методов класса "Эксперимент" доступ к данным и заполнение результатов эксперимента. Предусмотреть в классе метод, который обеспечивает заполнение результатов эксперимента из файла (15 баллов).

Для класса "Эксперимент" требуется определить методы, которые получают максимальные, минимальные и средние значения на базе этого эксперимента для заданных пользователем показателей (8 баллов).

Протестировать возможности созданных классов (10 баллов).

### 7.1. Основная литература:

1. Андрианова, А.А. Объектно-ориентированное программирование на C++. Часть 1. / А.А.Андрианова, Л.Н.Исмагилов, Т.М.Мухтарова. - Казань: Казанский федеральный университет, 2010.- 124 с.
2. Андрианова, А.А. Объектно-ориентированное программирование на C++. Часть 2. / А.А.Андрианова, Л.Н.Исмагилов, Т.М.Мухтарова. - Казань: Казанский федеральный университет, 2010.- 116 с.
3. Липачёв, Евгений Константинович. Технология программирования. Базовые конструкции C/C++: учебно-справочное пособие / Е. К. Липачёв; Казан. федер. ун-т. - Казань: [Казанский университет], 2012. - 139, [1] с
4. Липачёв Е.К. Технология программирования. Базовые конструкции C/C++. Учебно-справочное пособие // Казань: Казан. ун-т, 2012. - 142 с  
[http://kpfu.ru/publication?p\\_id=47437](http://kpfu.ru/publication?p_id=47437)
5. Немцова Т. И. Программирование на языке высокого уровня. Программ. на языке C++: Уч. пос. /Под ред. Л.Г.Гагариной - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 512 с. // <http://znanium.com/bookread.php?book=244875>

### 7.2. Дополнительная литература:

1. Объектно-ориентированный анализ и программирование [Текст: электронный ресурс] : конспект лекций / А. А. Андрианова, Т. М. Мухтарова ; М-во образования и науки РФ, ФГАОУ ВПО "Казан. (Приволж.) федер. ун-т", Ин-т вычисл. математики и информ. технологий .? Электронные данные (1 файл: 2,60 Мб) .? (Казань : Казанский федеральный университет, 2013) .? Загл. с экрана .? Для 2-го курса.  
Режим доступа: открытый .? <URL:[http://libweb.ksu.ru/ebooks/09-IVMIT/09\\_104\\_kl-000497.pdf](http://libweb.ksu.ru/ebooks/09-IVMIT/09_104_kl-000497.pdf)>.
2. Java : объектно-ориентированное программирование : для магистров и бакалавров : базовый курс по объектно-ориентированному программированию : [учебное пособие] / А. Н. Васильев .? Санкт-Петербург [и др.] : Питер, 2013 .? 395, [1] с. : ил. ; 24 .? (Учебное пособие) (Стандарт третьего поколения) .? По номеру записи 1878254 .? Библиогр.: с. 377 (11 назв.) .? Алф. указ. в конце кн. ? ISBN 978-5-496-00044-4 ((в обл.)) .? фрагмент книги.
3. Программирование на языке C# : учебно-методическое пособие / И. Л. Александрова, Д.Н. Тумаков ; Казан. федер. ун-т, Ин-т вычисл. мат. и информ. технологий .? Казань : [б. и.], 2011 .? 103 с. ; 21
4. Кальтин Н. Б. Основы программирования в Microsoft Visual C# 2010. ? СПб.: БХВ-Петербург, 2011. ? 364 с. ? Режим доступа: <http://www.znanium.com/bookread.php?book=351294>

### 7.3. Интернет-ресурсы:

Интернет-портал образовательных ресурсов КФУ - <http://www.kfu-elearning.ru/>

Интернет-портал образовательных ресурсов по ИТ - <http://www.intuit.ru>

Интернет-портал по программным средствам Microsoft - <http://msdn.microsoft.com/ru-ru/vcsharp/>

Интернет-портал со статьями по алгоритмике и программированию - <http://algolist.manual.ru/>

Электронная библиотека по техническим наукам - <http://techlibrary.ru>

### 8. Материально-техническое обеспечение дисциплины(модуля)

Освоение дисциплины "Объектно-ориентированный анализ и программирование" предполагает использование следующего материально-технического обеспечения:

Компьютерный класс, представляющий собой рабочее место преподавателя и не менее 15 рабочих мест студентов, включающих компьютерный стол, стул, персональный компьютер, лицензионное программное обеспечение. Каждый компьютер имеет широкополосный доступ в сеть Интернет. Все компьютеры подключены к корпоративной компьютерной сети КФУ и находятся в едином домене.

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе "ZNANIUM.COM", доступ к которой предоставлен студентам. ЭБС "ZNANIUM.COM" содержит произведения крупнейших российских учёных, руководителей государственных органов, преподавателей ведущих вузов страны, высококвалифицированных специалистов в различных сферах бизнеса. Фонд библиотеки сформирован с учетом всех изменений образовательных стандартов и включает учебники, учебные пособия, УМК, монографии, авторефераты, диссертации, энциклопедии, словари и справочники, законодательно-нормативные документы, специальные периодические издания и издания, выпускаемые издательствами вузов. В настоящее время ЭБС ZNANIUM.COM соответствует всем требованиям федеральных государственных образовательных стандартов высшего профессионального образования (ФГОС ВПО) нового поколения.

Компьютерные классы лаборатории малой вычислительной техники Института ВМ и ИТ, оснащенные мультимедийным оборудованием.

Программа составлена в соответствии с требованиями ФГОС ВПО и учебным планом по направлению 38.03.05 "Бизнес-информатика".

Автор(ы):

Андрианова А.А. \_\_\_\_\_

Мухтарова Т.М. \_\_\_\_\_

"\_\_" \_\_\_\_\_ 201\_\_ г.

Рецензент(ы):

Тагиров Р.Р. \_\_\_\_\_

Миссаров М.Д. \_\_\_\_\_

"\_\_" \_\_\_\_\_ 201\_\_ г.