Федеральное государственное бюджетное образовательное учреждение высшего образования
«РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА И ГОСУДАРСТВЕННОЙ СЛУЖБЫ ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»

Факультет информационных технологий и анализа данных
Кафедра системного анализа и информатики

УТВЕРЖДЕНА

решением кафедры системного анализа и информатики

Протокол от «03» сентября 2019 г.

№1

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1.Б.21 ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ АНАЛИЗ И ПРОГРАММИРОВАНИЕ

(индекс и наименование дисциплины (модуля), в соответствии с учебным планом)

по направлению подготовки
38.03.05 Бизнес-информатика
(код и наименование направления подготовки)

Цифровой бизнес
направленность (профиль)

Бакалавр
квалификация выпускника

Очная
форма обучения

набор 2020 г.

Москва, 2019 г.

.

Автор–составитель:

к.в.н., доцент Уманский Ю. Н.

(ученое звание, ученая степень, должность) *(Ф.И.О.)*

Заведующий кафедрой системного анализа и информатики

(наименование кафедры)

к.т.н., доцент Маруев С. А.

(ученая степень и(или) ученое звание) *(Ф.И.О.)*

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы…… 4
2. Объем и место дисциплины (модуля) в структуре образовательной программы ……... 4
3. Содержание и структура дисциплины (модуля)……... 5
4. Материалы текущего контроля успеваемости обучающихся и фонд оценочных средств промежуточной аттестации по дисциплине (модулю)……..……... 7
5. Методические указания для обучающихся по освоению дисциплины (модуля)……... 17
6. Учебная литература и ресурсы информационно-телекоммуникационной сети "Интернет", учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине (модулю)……..……. 19
   6.1. Основная литература……..……...…… 19
   6.2. Дополнительная литература……...……...…...…… 19
   6.3. Учебно-методическое обеспечение самостоятельной работы…… 19
   6.4. Нормативные правовые документы…….……..…….. 19
   6.5. Интернет-ресурсы…….……..……... 19
7. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы …...……...…..…….. 20

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения программы

Дисциплина «Объектно-ориентированный анализ и программирование» обеспечивает овладение следующими компетенциями:

Код компетенции	Наименование компетенции	Код этапа освоения компетенции	Наименование этапа освоения компетенции
ОПК-3.4	способность работать с компьютером как средством управления информацией, работать с информацией из различных источников, в том числе в глобальных компьютерных сетях	ОПК-3.4	Способен к самостоятельному использованию компьютера как средства управления информацией
ОПК-ОС-4	способность управлять ресурсами и сервисами информационных технологий (ИТ) и информационной средой (ИС)	ОПК-4.4	Способен управлять информационной средой

Менеджер продуктов в области информационных технологий. Приказ Минтруда России от 20.11.2014 N 915н (Зарегистрировано в Минюсте России 18.12.2014 N 35273).

В результате освоения дисциплины у студентов должны быть сформированы:

ОТФ/ТФ (при наличии профстандарта)	Код этапа освоения компетенции	Результаты обучения
С - Управление серией продуктов и группой их менеджеров. Заказ и анализ результатов технологических исследований в интересах серии продуктов (С/01.6) Разработка бизнес-планов, ценовой политики и стратегии развития серии продуктов (С/02.6) Заключение партнерских соглашений и развитие отношений с партнерами (С/01.3) Управление бюджетом серии продуктов (С/01.4) Управление группой менеджеров продуктов (С/01.5) Продвижение продуктов(С/016) Заказ и контроль выполнения программы проектов по созданию, развитию, выводу на рынок и продажам продуктов серии (С/01.7) Управление патентами на технологии, создаваемые в рамках продуктов (С/01.8) Разработка предложений по приобретению и продаже технологических, продуктовых и прочих интеллектуальных	ОПК-3.4 ОПК-4.4	Знать: З1 – направления решения типовых проблем разработки сложных программных систем с использованием технологий ООП З2 – принципы ООП как основы разработки сложных программных систем З3 – основные конструкции языка программирования С++ для работы с программными объектами
		Уметь: У1 – уметь разрабатывать алгоритм (проект) создаваемого приложения У2 – реализовывать разработанные алгоритмы с использованием средств языка программирования С++
		Владеть: В1 – типовыми приемами создания программного обеспечения с использованием технологии ООП

ОТФ/ТФ (при наличии профстандарта)	Код этапа освоения компетенции	Результаты обучения
активов и организаций (С/01.9)		

1. Объем и место дисциплины (модуля) в структуре ОП ВО

Дисциплина «Объектно-ориентированный анализ и программирование» имеет индекс Б1.Б.21, объем 144 академических часов, 4 з.е., изучается на 2 курсе в 4 семестре в соответствии с учебным планом. Количество академических часов, выделенных на контактную работу с преподавателем - 80 часов, на самостоятельную работу обучающихся - 28 часов; форма промежуточной аттестации – экзамен.

Для успешного освоения дисциплины «Объектно-ориентированный анализ и программирование» необходимы знания, умения и навыки в области информатики в объеме средней школы, а также курса «Высокоуровневые методы информатики и программирования», изучаемого в 3-м семестре.

Знания, умения и навыки по дисциплине «Объектно-ориентированный анализ и программирование» способствуют формированию общекультурных и профессиональных компетенций в рамках таких дисциплин как:
- «Управление жизненным циклом ИС»
- «Функциональное программирование и интеллектуальные системы»

2. Содержание и структура дисциплины (модуля)

№ п/п	Наименование тем (разделов)	Объем дисциплины, час.						Форма текущего контроля успеваемости[4], промежуточной аттестации
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СР	
			Л	ЛР	ПЗ	КСР		
Тема 1	Основные понятия технологии объектно-ориентированного программирования.	10	2		4		4	О
Тема 2	Определение собственных типов данных	10	2		4		4	О
Тема 3	Наследование классов и виртуальные функции.	12	2		6		4	Т1-3
Тема 4	Стандартная библиотека шаблонов.	14	2		8		4	О
Тема 5	Программирование для Windows с использованием библиотеки MFC.	14	2		8		4	КР
Тема 6	Рисование в окне	16	2		10		4	О
Тема 7	Работа с диалоговыми окнами и элементами	16	4		10		2	КР

№ п/п	Наименование тем (разделов)	Объем дисциплины, час.						Форма текущего контроля успеваемости[4], промежуточной аттестации
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СР	
			Л	ЛР	ПЗ	КСР		
	управления.							
Тема 8	Сохранение и печать документов.	16	4		10		2	КР
	Промежуточная аттестация	36						Экзамен
Всего по дисциплине		**144**	**20**		**60**		**28**	

** – при применении электронного обучения, дистанционных образовательных технологий в соответствии с учебным планом;*
*** – формы текущего контроля успеваемости: опрос (О), тестирование (Т), контрольная работа (КР), коллоквиум (К), эссе (Э), реферат (Р), диспут (Д) и др.*

Содержание дисциплины (модуля)

	Наименование тем (разделов)	Содержание тем (разделов)	Коды компетенций
Тема 1	Основные понятия технологии объектно-ориентированного программирования.	Теоретические основы объектно-ориентированного программирования. Необходимость перехода от процедурного программирования к объектному. Основные принципы объектно-ориентированного программирования. Этапы разработки программных систем с использованием ООП. Объектная декомпозиция, в т.ч. сложных предметных областей. Объекты и сообщения.	ПК-6
Тема 2	Определение собственных типов данных.	Структуры в С++. Использование указателей со структурами. Типы данных, объекты, классы и экземпляры. Операции с классами. Определение объектов класса. Конструкторы классов. Стандартный конструктор. Массивы объектов класса. Статические члены класса. Статические переменные-члены класса. Статические функции-члены класса. Указатели и ссылки на объекты классов. Ссылки на объекты класса. Деструкторы классов. Деструкторы и динамическое распределение памяти. Совместное использование памяти переменными. Определение объединений. Объединения в классах и структурах. Перегрузка операторов. Шаблоны классов. Создание объектов из шаблона класса. Шаблоны классов с несколькими параметрами. Шаблоны для объектов функций. Классы для работы со строками из “родной” библиотеки С++. Создание строковых объектов. Конкатенация строк. Обращение к строкам и их модификация. Сравнение строк. Поиск в строках.	ПК-6
Тема 3	Наследование классов и виртуальные функции.	Наследование в классах. Наследование классов от базового класса. Конструктор копирования в	ПК-6

		производном классе. Виртуальные функции. Приведение между типами классов. Вложенные классы.	
Тема 4	Стандартная библиотека шаблонов.	Стандартная библиотека шаблонов. Диапазон контейнеров STL. Последовательные контейнеры. Ассоциативные контейнеры. Дополнительные сведения об итераторах. Дополнительные сведения об объектах функций. Дополнительные сведения об алгоритмах. Лямбда-выражения. Шаблоны и лямбда-выражения. Оболочка лямбда-выражения.	ПК-6
Тема 5	Программирование для Windows с использованием библиотеки MFC.	Основы программирования для Windows. Программы Windows и операционная система. Программирование, управляемое событиями. Сообщения Windows. Интерфейс API Windows. Типы данных Windows. Нотация программ Windows. Структура программы Windows. Функция WinMain(). Организация программ Windows. Библиотека Microsoft Foundation Classes. Нотация библиотеки MFC. Структуризация программы MFC. Использование Windows Forms. Концепция "документ-представление" в библиотеке MFC. Создание приложений MFC. Работа с меню и панелями инструментов. Взаимодействие с Windows. Карты сообщений. Элементы меню. Создание и редактирование ресурсов меню. Добавление кнопок панели инструментов. Редактирование свойств кнопки панели инструментов. Добавление всплывающих подсказок.	ПК-6
Тема 6	Рисование в окне.	Основы рисования в окне. Механизм рисования в Visual C++. Практика рисования графики. Программирование для мыши. Рисование с помощью мыши. Захват сообщений мыши.	ПК-6
Тема 7	Работа с диалоговыми окнами и элементами управления.	Понятие диалоговых окон. Создание ресурса диалогового окна. Программирование для диалогового окна. Обработка сообщений переключателей. Завершение операций диалогового окна. Использование кнопки счетчика. Создание класса диалогового окна масштабирования. Отображение кнопки счетчика. Использование показателя масштаба. Масштабируемые режимы отображения. Установки размера документа. Установка режима отображения. Реализация прокрутки с масштабированием. Добавление панели состояния в обрамляющее окно. Использование окна списка. Создание элемента управления списком. Использование элемента управления - поле редактирования. Создание класса диалогового окна. Определение текстового элемента. Создание текстового элемента.	ПК-6
Тема 8	Сохранение и печать документов.	Сериализация документа. Сериализация в определении класса документа. Сериализация в реализации класса документа. Возможности классов на базе класса CObject. Применение сериализации. Запись изменений в документе. Сериализация документа. Сериализация классов элементов. Испытание сериализации. Печать документа. Процесс печати. Реализация многостраничной печати. Получение полного	ПК-6

		размера документа. Сохранение данных печати. Подготовка к печати. Очистка после печати. Подготовка контекста устройства. Печать документа. Получение печатного вывода документа.	

3. Материалы текущего контроля успеваемости обучающихся и фонд оценочных средств промежуточной аттестации по дисциплине (модулю)

4.1. Формы и методы текущего контроля успеваемости и промежуточной аттестации.

В ходе реализации дисциплины используются следующие методы текущего контроля успеваемости обучающихся:

-при проведении занятий лекционного типа: беседа (диалог) с обучающимися,

-при проведении занятий семинарского типа: домашние работы по темам практических заданий.

Контрольный опрос выполняется в письменном виде и включает задание в виде теста и задания по написанию фрагмента программного кода.

Типовой вариант теста по темам 1-3 приведен ниже (время выполнения 20 мин.):

1. В теории ООП под абстракцией понимают…
   - ☐ что-либо, утратившее определенность, конкретность, реальность;
   - ☐ совокупность существенных характеристик некоторого объекта, которые отличают его от всех других видов объектов;
   - ☐ совокупность существенных характеристик некоторого объекта, общих для одной группы объектов;
   - ☐ теоретическое обобщение.
2. Под инкапсуляцией в теории ООП понимают …
   - ☐ сочетание объединения всех свойств объекта, определяющих его поведение в единую абстракцию и ограничение доступа к реализации этих свойств;
   - ☐ сочетание объединения всех свойств объекта, определяющих его состояние и поведение в единую абстракцию и ограничение доступа к реализации этих свойств;
   - ☐ сочетание объединения всех свойств объекта, определяющих его состояние в единую абстракцию и ограничение доступа к реализации этих свойств;
   - ☐ сочетание объединения всех свойств объекта, определяющих его состояние и поведение в единую абстракцию и ограничение доступа к интерфейсу этих свойств.
3. Типизация, применительно к ООП, определяется как …
   - ☐ ограничение, накладываемое на свойства объектов и препятствующее взаимозаменяемости абстракций различных типов;
   - ☐ ограничение, накладываемое на свойства объектов и обеспечивающее легитимность взаимозаменяемости абстракций различных типов;
   - ☐ ограничение, накладываемое на свойства объектов и определяющее взаимозаменяемость абстракций различных типов.

4. По результатам анализа на первом этапе разработки программных систем с использованием ООП разрабатывают:
   - ☐ алгоритм;
   - ☐ блок-схему;
   - ☐ диаграмму классов;
   - ☐ диаграмму объектов;
   - ☐ диаграмму состояний интерфейса.

5. При выполнении логического проектирования на втором этапе разработки программных систем с использованием ООП:
   - ☐ принимаемые решения практически зависят от условий эксплуатации;
   - ☐ принимаемые решения практически не зависят от условий эксплуатации;

- ☐ принимаемые решения зависят только от логических условий программы.

6. Укажите этапы разработки программных систем с использованием ООП:
   - ☐ алгоритмизация;
   - ☐ анализ и уточнение спецификаций;
   - ☐ формализация;
   - ☐ проектирование;
   - ☐ разработка;
   - ☐ реализация;
   - ☐ отладка;
   - ☐ модификация;
   - ☐ сопровождение.
7. При объектной декомпозиции сложных систем используют:
   - ☐ диаграмму классов;
   - ☐ диаграмму объектов;
   - ☐ контекстную диаграмму классов.
8. Ассоциацией называют связь …
   - ☐ между разнотипными, не похожими друг на друга объектами;
   - ☐ между однотипными, похожими друг на друга объектами;
   - ☐ между логически связанными объектами.
9. Состояние объекта характеризуется …
   - ☐ набором конкретных значений его методов;
   - ☐ набором конкретных значений его свойств;
   - ☐ набором конкретных значений его событий.
10. Поведение объекта характеризуется:
    - ☐ набором конкретных значений его методов;
    - ☐ набором конкретных значений его свойств;
    - ☐ набором конкретных значений его событий.
11. Операция селекции предполагает …
    - ☐ выбор необходимых свойств объекта;
    - ☐ выбор необходимых методов объекта;
    - ☐ получение информации о состоянии объекта по запросу;
    - ☐ определение необходимых характеристик объектов.
12. Операция модификации предполагает …
    - ☐ обращение к объекту для изменения его состояния по всем составляющим;
    - ☐ обращение к объекту для изменения его состояния по всем или отдельным составляющим;
    - ☐ обращение к объекту для изменения его состояния по отдельным составляющим;
    - ☐ обращение к объекту для изменения его поведения по всем или отдельным составляющим.
13. Отношение использования предполагает:
    - ☐ передачу сообщения;
    - ☐ изменение свойства;
    - ☐ запуск метода;
    - ☐ реакцию на событие.
14. Класс в ООП определяется как …
    - ☐ структурный тип данных, который включает описание полей данных;
    - ☐ структурный тип данных, который включает описание полей данных, а также процедур и функций, работающих с этими полями;
    - ☐ структурный тип данных, который включает описание процедур и функций объекта.
15. Воздействие на объект выполняется посредством …

- ☐ изменения содержимого его полей;
- ☐ вызова его методов;
- ☐ изменения содержимого его полей или вызова его методов.

16. Методы создания и инициализации полей объекта называют …
    - ☐ конструкторы;
    - ☐ деструкторы;
    - ☐ селекторы.
17. Статическое создание объектов выполняется в процессе …
    - ☐ загрузки программы в оперативную память;
    - ☐ запуска программы на решение;
    - ☐ компиляции программы.
18. Механизм наследования позволяет классу-потомку использовать …
    - ☐ поля одного или нескольких родительских классов;
    - ☐ поля и методы одного или нескольких родительских классов;
    - ☐ методы одного или нескольких родительских классов;
    - ☐ поля одного и методы другого родительского класса.

Типовой вариант контрольной работы теста по теме 5 приведен ниже (время выполнения 4 академических часа):

Задание 1 (5 баллов). Определите структуру Sample, содержащую два целочисленных элемента данных. Напишите программу, которая объявит два объекта типа Sample с именами a и b. Установите значения для элементов данных, относящихся к объекту a, и затем убедитесь, что можете копировать значения объекта b простым присвоением.

Задание 2 (5 баллов). Определите класс CRecord с двумя закрытыми переменными-членами, которые хранят имя длиной до 14 символов и целочисленный элемент. Определите функцию-член getRecord() класса CRecord, которая установит значения переменных-членов, читая ввод с клавиатуры, и функцию-член putRecord(), выводящую значения переменных-членов. Реализуйте функцию getRecord() так, чтобы вызывающая ее программа могла обнаружить, когда вводится числовой элемент с нулевым значением. Проверьте класс CRecord с помощью функции main(), которая читает и выводит объекты CRecord до тех пор, пока не будет введен числовой элемент с нулевым значением.

Задание 3 (5 баллов). Реализуйте простой строковый класс на “родном” C++, который содержит char* и целочисленную длину как закрытые члены. Предоставьте конструктор, получающий аргумент типа const char*, и реализуйте конструктор копирования, оператор присвоения, а также функцию-деструктор. Удостоверьтесь, что класс работает. Легче всего для этого воспользоваться строковыми функциями из файла заголовка cstring.

Задание 4 (5 баллов). Пусть имеется описание класса комплексных чисел

```
class complex{
 double re, im;
};
```

Написать конструкторы, позволяющие создавать объект класса complex

а) по двум заданным параметрам

б) по заданному значению re (при этом считать im=0)

в) без параметров.

Задание 5 (5 баллов). Привести описание класса "обыкновенная дробь", члены-данные класса – числитель и знаменатель. При создании объектов этого класса возможно задание значений числителя и знаменателя или только числителя (знаменатель в этом случае считается равным 1). При создании дроби должно проводиться сокращение, то есть дроби ½ и 6/12 должны стать одинаковыми.

Типовой вариант контрольной работы теста по теме 7 приведен ниже (время выполнения 4 академических часа, все задания по 5 баллов):

1. Предположим, имеется приведенный ниже класс CBird, который вы хотите использовать как базовый для иерархии классов, описывающих птиц.

```
class CBird
{
        protected:
            int wingSpan;
            int eggSize;
            int airSpeed;
            int altitude;
            public:
                virtual void fly() { altitude = 100; }
    };
```

Разумно ли создавать класс CHawk (сокол), наследуя его от класса CBird (птица)?
Если да, то создайте его. Если нет, предложите свой класс CBird и на его основе класс CHawk .

2. Определить иерархию классов для описания сессии в университете. Базовый класс – event, содержит информацию о дате события, фамилию действующего лица и чисто виртуальную функцию print_res, печатающую информацию о событии; классы-наследники – test (зачет), exam (экзамен). В них должны быть определены дополнительные характеристики событий и метод print_res. В функции main определить сессию как массив указателей на события, проинициализировать элементы массива и распечатать информацию об экзаменах и зачетах.).

3. Привести пример программы, которая работает по-разному в зависимости от того, был ли объявлен деструктор виртуальным или нет.

4. Описать класс Date, для работы с датами (содержащий информацию о дне, месяце и годе). В классе должна быть описана функция set_date, которая устанавливает значения дня, месяца и года для даты, а при задании некорректных параметров генерирует исключение. Написать в функции main примеры вызовов написанной функции для демонстрации работы с корректными и с некорректными значениями. При возникновении исключения – печатать сообщение об ошибке.

5. Переопределить функции terminate и unexpected с помощью set_terminate и set_unexpected, чтобы функции печатали свое имя, а затем завершали работу программы. Продемонстрировать на примере работу переопределенных функций.

Типовой вариант контрольной работы теста по теме 8 приведен ниже (время выполнения 4 академических часа):

Задание 1 (5 баллов). Напишите на “родном” C++ программу для чтения некоторого текста из стандартного потока ввода, возможно, состоящего из нескольких строк ввода, сохраняя буквы текста в контейнере list<T>. Отсортируйте буквы в порядке возрастания и выведите их на экран.

Задание 2 (5 баллов). Написать программу, читающую последовательности символов со стандартного ввода. Признаком конца последовательности считать символ '$' (он в последовательность не входит). Обработать последовательность согласно одному из правил а) – е). При вводе пустой последовательности должно возникать исключение типа NullException. Типы исключений описать как классы-наследники exception, определить для них функцию what(). В функции main должна быть предусмотрена обработка возникших исключений. Информацию о типе исключения выдавать на экран.

Задание 3 (5 баллов). Распечатать сумму всех целых десятичных чисел, которые встретились в строке. Если эта сумма не помещается в тип int – сгенерировать исключение типа OverflowException. Если не встретилось ни одного числа – исключение типа NoDigits.

Задание 4 (5 баллов). Напишите программу на “родном” C++ для реализации телефонной книги, которая позволит вводить имя для извлечения одного или нескольких номеров либо вводить номер для получения имени.
Задание 5 (5 баллов). Описать шаблонную функцию max, которая возвращает значение максимального элемента для заданного массива целых чисел (int), длинных целых чисел (long), массива вещественных чисел (double) или массива строк (const char*). При работе со строками, максимальной считать ту строку, которая имеет наибольшую длину.

Шкала оценивания текущего контроля

10-бальная шкала	**Традиционная шкала**	**«Зачтено»/ «Не зачтено»**	**Определение**
10	Отлично	Зачтено	Полные, глубокие и систематические знания, знакомство с дополнительной литературой, полный и правильный ответ, творческий подход в понимании и изложении учебного материала, полное выполнение мероприятий текущего контроля.
9	Отлично	Зачтено	Полные, глубокие и систематические знания, полный и правильный ответ, полное выполнение мероприятий текущего контроля.
8	Отлично	Зачтено	Полные и систематические знания, отсутствие существенных неточностей в ответе, полное выполнение мероприятий текущего контроля.
7	Хорошо	Зачтено	Достаточно полные и систематические знания, отсутствие существенных неточностей в ответе, имеются погрешности при выполнении мероприятий текущего контроля.
6	Хорошо	Зачтено	Достаточно полные и систематические знания, отсутствие существенных неточностей в ответе, имеются погрешности при выполнении мероприятий текущего контроля.
5	Удовлетворительно	Зачтено	Знание основного учебного материала в объеме, необходимом для дальнейшей учебы и работы, имеются погрешности при выполнении мероприятий текущего контроля и при ответе.
4	Удовлетворительно	Зачтено	Знание основного учебного материала в минимальном объеме, необходимом для дальнейшей учебы и работы, имеются погрешности при выполнении мероприятий промежуточного контроля и при ответе.
3	Неудовлетворительно	Не зачтено	Имеются существенные погрешности при выполнении мероприятий текущего контроля, допущены существенные ошибки при ответе, необходима некоторая дополнительная работа.
2	Неудовлетворительно	Не зачтено	Имеются пробелы в знаниях по значительной части учебного материала, допущены существенные ошибки при ответе, необходима значительная дополнительная учебная работа.

10-бальная шкала	Традиционная шкала	«Зачтено»/ «Не зачтено»	Определение
1	Неудовлетворительно	Не зачтено	Не выполнены предусмотренные программой задания, не отработаны практические или лабораторные занятия, необходимы дополнительные занятия по соответствующей дисциплине.
0	Неудовлетворительно	Не зачтено	Нарушение академических норм (плагиат и т.п.)

4.3. Формы, методы (средства) промежуточной аттестации.

4.3.1. Для контроля усвоения данной дисциплины учебным планом предусмотрен экзамен *(в соответствии с учебным планом)*, который проводится в устной форме. Задания содержат вопросы, в которых необходимо использовать теоретические знания и практическое задание, демонстрирующие способность управления контентом предприятия и Интернет-ресурсов, процессами создания и использования информационных сервисов (контент-сервисов).

На экзамен выносятся основные вопросы, рассматриваемые в рамках всего курса. Основой для определения оценки служит объем и уровень усвоения студентами материала, предусмотренного программой данного курса и подведения итогов по результатам выполнения заданий текущего контроля успеваемости

ВОПРОСЫ
К ЭКЗАМЕНУ ПО ДИСЦИПЛИНЕ
"ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ АНАЛИЗ И ПРОГРАММИРОВАНИЕ"

1. Что определяет класс? Чем обличается класс от объекта?
2. Можно ли объявлять массив объектов? А массив классов?
3. Разрешается ли объявлять указатель на объект? А указатель на класс?
4. Допускается ли передавать объекты в качестве параметров, и какими способами? А возвращать как результат?
5. Как называется использование объекта одного класса в качестве поля другого класса?
6. Является ли структура классом? Чем класс отличается от структуры?
7. Какие ключевые слова в C++ обозначают класс?
8. Объясните принцип инкапсуляции.
9. Что такое композиция?
10. Для чего используются ключевые слова public и private?
11. Можно ли использовать ключевые слова public и private в структуре?
12. Существуют ли ограничения на использование public и private в классе? А в структуре?
13. Обязательно ли делать поля класса приватными?
14. Что такое метод? Как вызывается метод?
15. Может ли метод быть приватный?
16. Как определить метод непосредственно внутри класса? А вне класса? Чем эти определения отличаются?
17. Можно в методах присваивать параметрам значения по умолчанию?
18. Что обозначается ключевым словом this?
19. Зачем нужны константные методы? Чем отличается определение константного метода от обычного?
20. Может ли константный метод вызываться для объектов-переменных? А обычный метод — для объектов-констант?
21. Объясните принцип полиморфизма.

22. Сколько места в памяти занимает объект класса? Как это узнать?
23. Каков размер «пустого» объекта?
24. Влияют ли методы на размер объекта?
25. Одинаков ли размер класса и аналогичной структуры?
26. Какие операции нельзя перегружать? Как вы думаете, почему?
27. Можно ли перегружать операции для встроенных типов данных?
28. Можно ли при перегрузке изменить приоритет операции?
29. Можно ли определить новую операцию?
30. Перечислите особенности перегрузки операций как методов класса. Чем отличается перегрузка внешним образом от перегрузки как метода класса?
31. Какой результат должны возвращать операции с присваиванием?
32. Как различаются перегруженная префиксная и постфиксная операции инкремента и декремента?
33. Что означает выражение *this? В каких случаях оно используется?
34. Какие операции не рекомендуется перегружать как методы класса? Почему?
35. Какие операции разрешается перегружать только как методы класса?
36. Дайте определение дружественной функции. Как объявляется дружественная функция? А как определяется?
37. Дайте определение конструктора. Каково назначение конструктора? Перечислите отличия конструктора от метода.
38. Сколько конструкторов может быть в классе? Допускается ли перегрузка конструкторов? Какие виды конструкторов создаются по умолчанию?
39. Может ли конструктор быть приватным? Какие последствия влечет за собой объявление конструктора приватным?
40. Приведите несколько случаев, когда конструктор вызывается неявно.
41. Как проинициализировать динамическую переменную?
42. Как объявить константу в классе? Можно ли объявить дробную константу?
43. Каким образом разрешается инициализировать константные поля в классе?
44. В каком порядке инициализируются поля в классе? Совпадает ли этот порядок с порядком перечисления инициализаторов в списке инициализации конструктора?
45. Какие конструкции C++ разрешается использовать в списке инициализации качестве инициализирующих выражений?
46. Какой вид конструктора фактически является конструктором преобразования типов?
47. Для чего нужны функции преобразования? Как объявить такую функцию в классе?
48. Как запретить неявное преобразование типа, выполняемое конструктором инициализации?
49. Какие проблемы могут возникнуть при определении функций преобразования?
50. Для чего служит ключевое слово explicit?
51. Влияет ли наличие целочисленных констант-полей на размер класса?
52. Разрешается ли объявлять массив в качестве поля класса. Как присвоить элементам массива начальные значения?
53. Сколько операндов имеет операция индексирования []? Какой вид результата должна возвращать эта операция?
54. Для чего нужны статические поля в классе? Как они определяются?
55. Как объявить в классе и проинициализировать статический константный массив?
56. Что такое выравнивание и от чего оно зависит? Влияет ли выравнивание на размер класса?
57. Дайте определение контейнера.
58. Какие виды встроенных контейнеров в C++ вы знаете?
59. Какие виды доступа к элементам контейнера вам известны?
60. Чем отличается прямой доступ от ассоциативного?

61. Перечислите операции, которые обычно реализуются для последовательного доступа к элементам контейнера.
62. Дайте определение итератора.
63. Можно ли реализовать последовательный доступ без итератора? В чем преимущества реализации последовательного доступа с помощью итератора?
64. Что играет роль итератора для массивов C++?
65. Что такое деструктор? Может ли деструктор иметь параметры?
66. Почему для классов-контейнеров деструктор надо писать явным образом?
67. Допускается ли перегрузка деструкторов?
68. Какое копирование осуществляет стандартный конструктор копирования?
69. Чем отличается копирование от присваивания?
70. Объясните, почему в операции присваивания требуется проверка присваивания самому себе?
71. Можно ли в качестве операции индексирования использовать операцию вызова функции ()? В чем ее преимущества перед операцией []?
72. Каков порядок вызова конструкторов? А деструкторов?
73. Можно ли в производном классе объявлять новые поля? А методы?
74. Если имя нового поля совпадает с именем унаследованного, то каким образом разрешить конфликт имен?
75. Что происходит, если имя метода-наследника совпадает с именем базового метода?
76. Объясните, зачем нужны виртуальные функции.
77. Чем «раннее» связывание отличается от «позднего»?
78. Может ли виртуальная функция быть дружественной функцией класса?
79. Наследуются ли виртуальные функции?
80. Может ли конструктор быть виртуальным? А деструктор?
81. Как объявляется «чистая» виртуальная функция?
82. Дайте определение абстрактного класса.
83. Наследуются ли чистые виртуальные функции?
84. Можно ли объявить деструктор чисто виртуальным?
85. В чем заключается отличие аргумента «файл» от <файл> в директиве #include?
86. Объясните, что такое «страж» включения и зачем он нужен.
87. Можно ли использовать слово extern при объявлении функций?
88. Чем отличаются именованные и неименованные пространства имен?
89. Могут ли пространства имен быть вложенными?
90. Чем отличается статический метод от обычного?
91. Какие методы класса не могут быть статическими?
92. Какие применения статических полей вы можете привести? А каким образом применяются статические методы?
93. Для чего предназначены шаблоны?
94. Какие виды шаблонов в C++ вы знаете?
95. Объясните назначение ключевого слова typename.
96. В каких единицах выдает результат операция sizeof? Какие типы данных имеют размер 1?
97. В чем главное отличие объектно-ориентированного механизма new/delete от механизма malloc()/free()?
98. Сколько существует форм new/delete? В чем их отличие?
99. Какие типы являются POD-типами? Чем отличается работа механизма new/delete с POD-объектами и nonPOD-объектами?
100. Какие функции выполняет обработчик new?
101. Можно ли реализовать собственный обработчик new и «прицепить» его к механизму new/delete?
102. Что такое «функтор»? Приведите пример функционального класса.

103. Какими способами функтор вызывается?
104. Можно ли с помощью алгоритма for_each() изменить элементы контейнера?

Показатели и критерии оценивания компетенций по этапам их формирования

Этап освоения компетенции	**Показатель оценивания** *Что делает обучающийся (какие действия способен выполнить), подтверждая этап освоения компетенции*	**Критерий оценивания** *Как (с каким качеством) выполняется действие. Соответствует оценке «отлично» в шкале оценивания в РПД.*	**Оценка (баллы)**
ОПК-3.4 Способен к самостоятельному использованию компьютера как средства управления информацией	Информационные продукты (документы) содержащие алгоритмы решения базовых классов задач (классификации, регрессии, кластеризации, ассоциации) бизнес-анализа данных организаций и предприятий	Разработанный алгоритм решения является оптимальным, результаты применимы для поддержки принятия решений	Тест
4 этап (код этапа: ОПК ОС-4.4) Способен управлять информационной средой	Способен использовать навыки применения методов интеллектуального и топологического анализа данных и программирования для решения экономических задач и управления информационной средой.	Определено оптимальное количество необходимых для разработки информационной среды ресурсов. Определены все возможные ограничения, существующие в рамках реализации информационной среды.	Промежуточная аттестация тест

Таблица 5.
Шкала оценивания по темам дисциплины

Наименование темы (раздела)	**Показатели оценивания**	**Критерии оценивания**	**Оценка (баллы)**
1. Основные понятия технологии объектно-ориентированного программирования.			
2. Определение собственных			

типов данных.			
3. Наследование классов и виртуальные функции.	Т1-3 (18 баллов)	< 9 баллов от 9 до 12 баллов от 13 до-16 баллов от 17 до 18 баллов	2 3 4 5
4. Стандартная библиотека шаблонов.			
5. Программирование для Windows с использованием библиотеки MFC.	КР (25 баллов)	< 5 баллов от 6 до 13 баллов от 14 до-20 баллов от 21 до 25 баллов	2 3 4 5
6. Рисование в окне.			
7. Работа с диалоговыми окнами и элементами управления.	КР (25 баллов)	< 5 баллов от 6 до 13 баллов от 14 до-20 баллов от 21 до 25 баллов	2 3 4 5
8. Сохранение и печать документов.	КР (25 баллов)	< 5 баллов от 6 до 13 баллов от 14 до-20 баллов от 21 до 25 баллов	2 3 4 5
Экзамен	1. Результаты оценки 1-го вопроса билета 2. Результаты оценки 2-го вопроса билета 3. Результаты оценки 3-го практического задания (разработка программы)	Менее 12 баллов 2 13-16 баллов 3 17-22 баллов 4 23-25 баллов 5 Менее 12 баллов 2 13-16 баллов 3 17-22 баллов 4 23-25 баллов 5 Менее 25 баллов 2 26-35 баллов 3 36-44 баллов 4 45-50 баллов 5	Сумма баллов по 3-м вопросам.

Порядок формирования оценок по дисциплине:

Для формирования итоговой оценки по дисциплине принята следующая система весов:

20% результирующей оценки – оценка за работу на семинарских занятиях;

40% результирующей оценки – взвешенная сумма оценок за контрольные мероприятия;

40% результирующей оценки – оценка за итоговый (или промежуточный контроль).

Таким образом, 60% результирующей оценки – это накопительная оценка и 40% – это оценка за итоговый (или промежуточный контроль).

Результирующая оценка рассчитывается с помощью взвешенной суммы накопительной оценки и оценки за экзамен (или зачет).

Накопительная оценка рассчитывается с помощью взвешенной суммы оценок за отдельные формы текущего контроля. К формам текущего контроля относятся контрольные мероприятия (контрольные работы, эссе, коллоквиумы и пр.), которые определены учебным планом.

Формулы расчета оценок:

$O_{текущая} = n_1 \cdot O_{к/р}$

где O_i – оценки за контрольные мероприятия (эссе, контрольная работа, реферат и пр.)

n_i – вес контрольных мероприятий (определяются преподавателем и $\sum n_i=1$ или 100%), при этом

Веса по контрольным мероприятиям:

n_1 = 100% - контрольная работа.

$O_{накопительная} = k_1 \cdot O_{текущая} + k_2 \cdot O_{аудиторная}$

где k_i – вес текущей и аудиторной оценки, при этом k_1=2/3, k_2=1/3

$O_{результирующая} = q_1 \cdot O_{накопительная} + q_2 \cdot O_{итог.контроль}$

где q_i – вес накопительной оценки и оценки за итоговый контроль, при этом q_1=0,6, q_2=0,4

4.4. Методические материалы по проведению промежуточной аттестации

Экзамен проводится в соответствии с графиком учебного процесса учетом проведения мониторинга уровня освоения компетенции по результатам выполнения самостоятельных заданий. Оценивание осуществляется в соответствии со шкалой оценивания. Студентам, не выполнившим домашние задания и (или) контрольные задания по уважительным причинам, предоставляется возможность их выполнения и сдачи.

4. Методические указания для обучающихся по освоению дисциплины

Самостоятельная работа студентов по дисциплине «Высокоуровневые методы информатики и программирования» состоит в

1) выполнении самостоятельных работ на компьютерах с установленной бесплатной версией среды разработки Visual C++;

2) подготовке к опросу по пройденному материалу.

Самостоятельная работа предполагает получение индивидуального задания каждым студентом по пройденному материалу. Для начала работы с заданием необходимо установить бесплатное (для студентов) программное обеспечение в соответствии с инструкциями компании Microsoft и приступить к самостоятельной работе. Данный задание оценивается как по срокам выполнения, так и по количеству и качеству допущенных ошибок.

Подготовка домашнего задания является основной формой самостоятельной работы студентов. Студенты получают задание на каждом практическом занятии и с той же регулярностью отчитываются о его выполнении. Бóльшая часть заданий имеет письменный характер. В связи с этим студенту рекомендуется завести отдельную тетрадь для самостоятельной работы, которая предъявляется преподавателю по его требованию. Индивидуальное задание назначается студенту в целях совершенствования актуальных для него навыков или для наилучшего усвоения отдельных тем дисциплины.

Подготовка докладов и сообщений может широко использоваться студентами при подготовке к практическим занятиям. Данный вид самостоятельной работы рассматривается как вспомогательный. В то же время темы выступлений на занятиях могут быть развернуты в темы студенческих научных исследований и стать основой для участия в студенческих научно-практических конференциях, олимпиадах, конкурсах студенческих научных работ.

1. Советы по планированию и организации времени, необходимого для изучения дисциплины. Рекомендуется следующим образом организовать время, необходимое для изучения дисциплины: Изучение конспекта лекции в тот же день после лекции – 10-15 минут. Повторение лекции за день перед следующей лекцией – 10-15 минут. Изучение теоретического материала по учебнику и конспекту – 1 час в неделю. Подготовка к практическому занятию – 1 час. Тогда общие затраты времени на освоение курса студентами составят около 2,5 часа в неделю.

2. Описание последовательности действий студента («сценарий изучения дисциплины»). Следует внимательно слушать и конспектировать материал, излагаемый на аудиторных занятиях. Для его понимания и качественного усвоения рекомендуется следующая последовательность действий: 1. После окончания учебных занятий для закрепления материала просмотреть и обдумать текст лекции, прослушанной сегодня, разобрать рассмотренные примеры (10-15 минут). 2. При подготовке к лекции следующего дня повторить текст предыдущей лекции, подумать о том, какая может быть следующая тема (10-15 минут). 3. В течение недели выбрать время для работы с

литературой в библиотеке и для решения задач (по 1 часу). 4. При подготовке к практическим занятиям повторить основные понятия и разобрать примеры на компьютере. Решая упражнение или задачу, – предварительно понять, какой теоретический материал нужно использовать. Наметить план решения, попробовать на его основе решить 1-2 аналогичные задачи.

4. Рекомендации по работе с литературой. Рекомендуется, кроме «заучивания» материала, добиться понимания изучаемой темы дисциплины. С этой целью после прочтения очередной главы желательно выполнить несколько простых упражнений на соответствующую тему. Кроме того, очень полезно мысленно задать себе и попробовать ответить на следующие вопросы: о чем эта глава, какие новые понятия в ней введены, каков их смысл.

5. Советы по подготовке к зачету. Дополнительно к изучению конспектов лекций необходимо пользоваться учебниками по дисциплине. Вместо «заучивания» материала важно добиться понимания изучаемых тем дисциплины. При подготовке к нужно освоить теорию: разобрать определения всех понятий структурного программирования, рассмотреть примеры и самостоятельно решить несколько типовых задач из каждой темы. При решении задач всегда необходимо комментировать свои действия и не забывать о содержательной интерпретации.

6. Указания по организации работы с контрольно-измерительными материалами. При выполнении домашних заданий и подготовке к контрольной работе необходимо сначала прочитать теорию и изучить примеры по каждой теме. Решая конкретную задачу, предварительно следует понять, что требуется от Вас в данном случае, какой теоретический материал нужно использовать, наметить общую схему решения. Если задача решается «по образцу» рассмотренного на практическом занятии или в методическом пособии примера, то желательно после этого обдумать процесс решения и попробовать решить аналогичную задачу самостоятельно.

5. Учебная литература и ресурсы информационно-телекоммуникационной сети «Интернет», включая перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1 Основная литература.

1. Кобзарь А.И. Прикладная математическая статистика. – М.: Физматлит, 2012
2. Казанский, А. А. Объектно-ориентированный анализ и программирование на visual basic 2013 : учебник для среднего профессионального образования / А. А. Казанский. — Москва : Издательство Юрайт, 2019. — 290 с. — (Профессиональное образование). — ISBN 978-5-534-03833-0. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://www-biblio-online-ru.ezproxy.ranepa.ru:2443/bcode/437247

6.2. Дополнительная литература.

1. Кулаичев А.П. Методы и средства комплексного анализа данных. - М.: Форум – Инфра-М, 2006.

6.3. Учебно-методическое обеспечение самостоятельной работы.

1. Ласковец С.В. Методология научного творчества [Электронный ресурс]: Учебное пособие. – Москва : Евразийский открытый институт, 2010. – 32 с. – URL: http://www.biblioclub.ru/90384_Metodologiya_nauchnogo_tvorchestva_Uchebnoe_posobie.html
2. Радаев В.В. Как организовать и представить исследовательский проект. 75 простых правил. – Москва : ГУ-ВШЭ : Инфра-М, 2001. – 203 с.

3. Панкратов В.Н. Искусство управлять собой: Практическое руководство. – Москва: Издательство института психотерапии, 2001. – 256 с.
4. ПОЛОЖЕНИЕ об организации самостоятельной работы студентов федерального государственного бюджетного образовательного учреждения высшего образования «Российская академия народного хозяйства и государственной службы при Президенте Российской Федерации» (в ред. приказа РАНХиГС от 11.05.2016 г. № 01-2211)
5. ПОРЯДОК освоения в РАНХиГС факультативных и элективных дисциплин (модулей) образовательных программ высшего образования - программ бакалавриата, программ специалитета, программ магистратуры. Приложение к приказу от 26 июля 2016 г. № 02-417.

6.4. Нормативные правовые документы.

1. Федеральный закон от 29.12.2012 N 273-ФЗ (ред. от 13.07.2015) "Об образовании в Российской Федерации" (с изм. и доп.).
2. Федеральный государственный образовательный стандарт высшего профессионального образования по направлению подготовки 38.03.05 Бизнес-информатика (квалификация (степень) «бакалавр»), утвержденный приказом Министерства образования и науки Российской Федерации от «11»августа 2016 г. № 1002.
3. Нормативно-методические документы Минобрнауки России.
4. Устав Федерального государственного бюджетного образовательного учреждения высшего профессионального образования «Российская академия народного хозяйства и государственной службы при Президенте РФ».

6.5. Интернет-ресурсы, справочные системы.

1. AnnualReviews [Электронный ресурс]. – URL: http://arjournals.annualreviews.org/.
2. EbscoHost [Электронный ресурс]. – URL: http://www.ebscohost.com/.
3. e-Library.ru [Электронный ресурс]: Научная электронная библиотека. – URL: http://elibrary.ru/.
4. Научная электронная библиотека «КиберЛенинка» [Электронный ресурс]. – URL: http://cyberleninka.ru//
5. ProQuest [Электронный ресурс]. – URL: http://search.proquest.com/index.
6. http://www.hr-training.net
7. http://www.lseptember.ru
8. http://www.tolerance.ru
9. Словари и энциклопедии на Академике [Электронный ресурс] // Академик. – URL: http://dic.academic.ru.
10. Университетская библиотека online [Электронный ресурс]. – URL: http://biblioclub.ru/.
11. Консультант Плюс http://www.consultant.ru/law/hotdocs/t7/

6. Материально-техническое и программное обеспечение дисциплины (модуля)

7.1. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Для реализации данной дисциплины (модуля), необходимы специализированные компьютерные аудитории для проведения всех видов контактной и самостоятельной работы. Аудитории должны быть оборудованы компьютерами в соответствии с минимальными техническими требованиями. Количество рабочих мест обучаемых должно быть не менее количества студентов в учебной группе. При использовании виртуальных машин должен быть единый защищенный сетевой ресурс, на котором обучаемые смогут сохранять результаты своей работы. В обязательном порядке в

аудитории должна присутствовать проекционная аппаратура, обеспечивающая как показ презентаций по теме занятий, так и демонстрацию работы преподавателя в среде разработки в реальном режиме времени. Оборудование класса должно обеспечивать выход преподавателя и обучаемых в глобальную сеть Интернет для выполнения учебных занятий. К обязательному программному обеспечению для поддержки образовательного процесса необходимо отнести: MS Excel.

7.2. Информационные технологии, используемые при осуществлении образовательного процесса по дисциплине (модулю)

При осуществлении образовательного процесса применяются информационные технологии, необходимые для подготовки презентационных материалов и материалов к занятиям (компьютеры с программным обеспечением для создания и показа презентаций, с доступом в сеть «Интернет», поисковые системы и справочные, профессиональные ресурсы в сети «Интернет»).

Содержание дисциплины размещено на сайте информационно-коммуникационной сети Интернет: ranepa.ru/.

7.3. Необходимое программное обеспечение

Для подготовки презентаций и их демонстрации необходима программа Impress из свободного пакета офисных приложений OpenOffice (или иной аналог с коммерческой или свободной лицензией).

Для контактной и самостоятельной работы используются мультимедийные комплексы, электронные учебники и учебные пособия, адаптированные к ограничениям здоровья обучающихся. Информационные средства обучения: электронные учебники, учебные фильмы по тематике дисциплины, презентации, интерактивные учебные и наглядные пособия, технические средства предъявления информации (многофункциональный мультимедийный комплекс) и контроля знаний (тестовые системы).