

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Рукович Александр Владимирович

Должность: Директор

Дата подписания: 30.05.2025 14:33:21

Уникальный программный ключ:

f45eb7c44954саас05ea7d4f32eb8d7d6b3cb96ae6d9b4bda094afddaffb705f

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«СЕВЕРО-ВОСТОЧНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ ИМЕНИ М.К. АММОСОВА»
Технический институт (филиал) ФГАОУ ВО «СВФУ» в г. Нерюнгри

Кафедра Математики и информатики

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Б1.В.03 Объектно-ориентированное программирование

для программы бакалавриата
по направлению подготовки 01.03.02 «Прикладная математика и информатика»
Направленность (профиль) программы: Системное программирование и компьютерные
технологии

Форма обучения: очная

Нерюнгри 2023

УТВЕРЖДЕНО на заседании

выпускающей кафедры

Мии

« 05 » 05 20 23 г., протокол № 10

Заведующий кафедрой

« 05 » 05 20 23 г.

Самохина В.М.

УТВЕРЖДЕНО на заседании

обеспечивающей кафедры

Мии

« 05 » 05 20 23 г., протокол № 10

Заведующий кафедрой

« 05 » 05 20 23 г.

Самохина В.М.

СОГЛАСОВАНО:

Эксперты¹:

Юданова В.В., ст. преподаватель кафедры Мии

Ф.И.О., должность, организация

подпись

Самохина В.М., к.п.н, доцент кафедры Мии

Ф.И.О., должность, организация

подпись

СОСТАВИТЕЛЬ (И):

Похорукова М.Ю., к.т.н., доцент кафедры Мии

Ф.И.О., должность, организация

подпись

¹ Эксперт первый: со стороны выпускающей кафедры (или работодатель). Эксперт второй: со стороны обеспечивающей кафедры.

Паспорт фонда оценочных средств
Б1.В.03 Объектно-ориентированное программирование

№	Контролируемые разделы (темы)	Код контролируемой компетенции (или ее части)	Требования к уровню усвоения компетенции	Наименование оценочного средства
1.	Объектно-ориентированное программирование. Язык C++	ПК-2.2: Способен приобретать новые и использовать существующие профессиональные знания в области типовых решений, библиотек программных модулей, используемых при разработке программного обеспечения	знать: основы архитектуры операционных систем семейства Windows, принципы объектно-ориентированного программирования. уметь: применять на практике технологии программирования, разрабатывать прикладные программные продукты с использованием API операционных систем семейства Windows и принципов объектно-ориентированного программирования. владеть: навыками проектирования программ в соответствии с объектно-ориентированным подходом.	лабораторные занятия, самостоятельные работы, контрольная работа, аттестационная работа, экзамен
2.	Классы и объекты C++	ПК-2.3: Способен использовать принципы построения архитектуры программного обеспечения и виды архитектур программного обеспечения ПК-3.1: Способен выполнять формализацию и алгоритмизацию поставленных задач в соответствии с требованиями технического задания		
3.	Перегрузка операций в ООП. Наследование в ООП.	ПК-3.2: Способен написать программный код с использованием языков программирования, использовать выбранную среду программирования и средства системы управления базами данных, стандартные библиотеки языка программирования		
4.	Шаблоны функций. Шаблоны классов. Паттерны проектирования.	ПК-3.3: Способен применять методы и приемы отладки программного кода, интерпретировать сообщения об ошибках, применять современные компиляторы, отладчики программного кода		

Программа экзамена

Экзамен проводится в форме собеседования по экзаменационным билетам. Программа экзамена включает в себя 2 теоретических вопроса и 1 практическое задание, направленное на выявление уровня сформированности компетенций ПК-2, ПК-3.

Вопросы к экзамену (4 семестр):

1. Объектно-ориентированное программирование. Язык C++.
2. Инкапсуляция.
3. Наследование.
4. Полиморфизм.
5. Ключевые слова static, const.
6. Классы в ООП.
7. Спецификаторы public, private, protected.
8. Глобальные и локальные классы.
9. Конструкторы классов и их свойства.
10. Конструкторы по умолчанию.
11. Конструктор копирования.
12. Статические элементы класса.
13. Дружественные функции и классы.
14. Деструкторы.

Вопросы к экзамену (5 семестр):

1. Перегрузка операций в ООП.
2. Перегрузка унарных, бинарных операций
3. Перегрузка операции присваивания, приведения типа.
4. Особенности работы операторов new и delete.
5. Наследование в ООП.
6. Виды наследования.
7. Простое наследование.
8. Виртуальные методы, виртуальный деструктор.
9. Абстрактные классы.
10. Множественное наследование в ООП.
11. Альтернатива наследованию.
12. Отличия структур и объединений от классов
13. Шаблоны функций.
14. Шаблоны классов.
15. Паттерны проектирования.

Типовое практическое задание

Опишите класс для хранения имени, места работы и возраста сотрудника с двумя конструкторами: без аргументов и с аргументами для инициализации указанных полей.

Критерии оценки:

Компетенции	Характеристика ответа на теоретический вопрос / выполнения практического задания	Количество набранных баллов
ПК-2,3	Дан полный, развернутый ответ на поставленный вопрос, показана совокупность осознанных знаний по дисциплине, доказательно раскрыты основные положения вопросов; в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий. Знание по предмету демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей. Могут быть допущены недочеты в определении понятий, исправленные студентом самостоятельно в процессе ответа.	10 б.

	Дан полный, развернутый ответ на поставленный вопрос, показано умение выделить существенные и несущественные признаки, причинно-следственные связи. Ответ четко структурирован, логичен. Могут быть допущены 2-3 неточности или незначительные ошибки, исправленные студентом с помощью преподавателя.	8 б.
	Дан недостаточно полный и недостаточно развернутый ответ. Логика и последовательность изложения имеют нарушения. Допущены ошибки в раскрытии понятий, употреблении терминов. Студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи. В ответе отсутствуют выводы. Умение раскрыть значение обобщенных знаний не показано.	5 б.
	Ответ представляет собой разрозненные знания с существенными ошибками по вопросу. Присутствуют фрагментарность, нелогичность изложения. Студент не осознает связь обсуждаемого вопроса по билету с другими объектами дисциплины. Отсутствуют выводы, конкретизация и доказательность изложения. Дополнительные и уточняющие вопросы преподавателя не приводят к коррекции ответа студента. <i>или</i> Ответ на вопрос полностью отсутствует <i>или</i> Отказ от ответа	0 б.
	Практическое задание выполнено верно, отсутствуют ошибки различных типов. Могут быть допущены недочеты в определении понятий, исправленные студентом самостоятельно в процессе ответа.	10 б.
	Практическое задание выполнено в полном объеме. Допущена незначительная ошибка.	8 б.
	Допущены несколько незначительных ошибок различных типов.	5 б.
	Допущены значительные ошибки. Дополнительные и уточняющие вопросы преподавателя не приводят к коррекции ответа студента. <i>или</i> Выполнение практического задания полностью неверно, отсутствует	0 б.

Лабораторные работы

В период освоения дисциплины студенты посещают лекционные занятия, самостоятельно изучают дополнительный теоретический материал к лабораторным занятиям. Критериями оценки работы на лабораторных занятиях является: полнота и правильность выполненного задания; степень осознанности, понимания изученного; оформление задания.

Содержание отчета.

1. Титульный лист: название дисциплины; номер и наименование работы; фамилия, имя, отчество студента; дата выполнения.

2. Задание.

3. Листинг программы с выполнением задания.

4. Результаты работы программы.

5. Вывод по работе, сформулированные из цели.

Темы лабораторных работ

Тема 1. Объектно-ориентированное программирование. Язык C++

Практические задания для самостоятельного решения

Разработать классы для описанных ниже объектов. Включить в класс методы `set (...)`, `get (...)`, `show (...)`. Определить другие методы.

1. **Student:** Фамилия, Имя, Отчество, Дата рождения, Адрес, Телефон, Факультет, Курс. Создать массив объектов. Вывести:

а) список студентов заданного факультета; б) списки студентов для каждого факультета и курса;

в) список студентов, родившихся после заданного года.

2. **Abiturient:** Фамилия, Имя, Отчество, Адрес, Оценки. Создать массив объектов. Вывести:

а) список абитуриентов, имеющих неудовлетворительные оценки; б) список абитуриентов, сумма баллов у которых не меньше заданной;

в) выбрать N абитуриентов, имеющих самую высокую сумму баллов, и список абитуриентов, имеющих полупроходной балл.

3. **Aeroflot:** Пункт назначения, Номер рейса, Тип самолета, Время вылета, Дни недели. Создать массив объектов. Вывести:

а) список рейсов для заданного пункта назначения; б) список рейсов для заданного дня недели;

в) список рейсов для заданного дня недели, время вылета для которых больше заданного.

4. **Book:** Автор, Название, Издательство, Год, Количество страниц. Создать массив объектов. Вывести:

а) список книг заданного автора; б) список книг, выпущенных заданным издательством;

в) список книг, выпущенных после заданного года.

5. **Worker:** Фамилия и инициалы, Должность, Год поступления на работу, Зарплата. Создать массив объектов. Вывести:

а) список работников, стаж работы которых на данном предприятии превышает заданное число лет; б) список работников, зарплата которых больше заданной;

в) список работников, занимающих заданную должность.

6. **Train:** Пункт назначения, Номер поезда, Время отправления, Число общих мест, Купейных, Плацкартных. Создать массив объектов. Вывести:

а) список поездов, следующих до заданного пункта назначения; б) список поездов, следующих до заданного пункта назначения и отправляющихся после заданного часа; в) список поездов, отправляющихся до заданного пункта назначения и имеющих общие места.

7. **Product:** Наименование, Производитель, Цена, Срок хранения, Количество. Создать массив объектов. Вывести:

а) список товаров для заданного наименования; б) список товаров для заданного наименования, цена которых не превышает указанной;

в) список товаров, срок хранения которых больше заданного.

8. Patient: Фамилия, Имя, Отчество, Адрес, Номер медицинской карты, Диагноз. Создать массив объектов. Вывести:

а) список пациентов, имеющих данный диагноз; б) список пациентов, номер медицинской карты которых находится в заданном интервале.

9. Person: Фамилия, Имя, Отчество, Адрес, Пол, Образование, Год рождения. Создать массив объектов. Вывести:

а) список граждан, возраст которых превышает заданный; б) список граждан с высшим образованием; в) список граждан мужского пола.

Написать программу, в которой создаются и разрушаются объекты, определенного пользователем класса. Выполнить исследование вызовов конструкторов и деструкторов.

Варианты заданий

1. АДРЕС имя – char* улица – char* номер дома – int	6. КОРАБЛЬ имя – char* водоизмещение – int тип – char*
2. СТРАНА имя – char* форма правления – char* площадь – float	7. КВИТАНЦИЯ номер – int дата – int сумма – float
3. БИБЛИОТЕКА имя – char* автор – char* стоимость – float	8. КАДРЫ имя – char* номер цеха – int разряд – int
4. ЭКЗАМЕН имя студента – char* дата – int оценка – int	9. СЛУЖАЩИЙ имя – char* возраст – int рабочий стаж – int
5. АВТОМОБИЛЬ марка – char* мощность – int стоимость – float	10. ЖИВОТНОЕ имя – char* класс – char* средний вес – int

Тема 2. Классы и объекты C++

Практические задания для самостоятельного решения

Разработать перечисленные ниже классы. При разработке каждого класса возможны два варианта решения: а) данные-члены класса представляют собой переменные и массивы фиксированной размерности; б) память для данных-членов класса выделяется динамически.

1. «Комплексное число» – **Complex**. Класс должен содержать несколько конструкторов и операции для сложения комплексных чисел.

2. Определить класс «Дробь» – **Fraction** в виде пары (m, n) . Класс должен содержать несколько конструкторов. Перегрузить операции сложения, вычитания, присваивания дробей.

3. Разработать класс «Вектор» – **Vector** размерности n . Определить несколько конструкторов, в том числе конструктор копирования. Реализовать методы для вычисления модуля вектора, сложения векторов.

4. Определить класс «**Квадратная матрица**» – **Matrix**. Класс должен содержать несколько конструкторов, в том числе конструктор копирования. Перегрузить операции сложения, вычитания. В головной программе вывести результат.

5. Определить класс «**Строка**» – **String** длины n . Написать несколько конструкторов, в том числе конструктор копирования. Реализовать методы для выполнения конкатенации строк.

6. Определить класс «**Матрица**» – **Matrix**. Класс должен содержать несколько конструкторов, в том числе конструктор копирования. Реализовать методы для сложения, вычитания. Перегрузить операции сложения, вычитания и присваивания для данного класса. В головной программе вывести результат.

Тема 3. Перегрузка операций в ООП. Наследование в ООП.

Практические задания для самостоятельного решения

1. Создать абстрактный базовый класс с виртуальной функцией - площадь. Создать производные классы: прямоугольник, круг со своими функциями площади. Для проверки определить массив ссылок на абстрактный класс, которым присваиваются адреса различных объектов.

2. Создать базовый класс - Предок, у которого есть имя. определить виртуальную функцию печати. Создать производный класс Ребенок, у которого функция печати дополнительно выводит имя. Создать производный класс от последнего класса - Внук, у которого есть отчество. Написать свою функцию печати.

3. Создать абстрактный базовый класс с виртуальной функцией - корни уравнения. Создать производные классы: класс линейных уравнений и класс квадратных уравнений. Определить функцию вычисления корней уравнений.

4. Создать абстрактный базовый класс с виртуальной функцией - объем. Создать производные классы: параллелепипед, пирамида функциями объема. Для проверки определить массив ссылок на абстрактный класс, которым присваиваются адреса различных объектов. Объем параллелепипеда - $V=xyz$ (x, y, z - стороны), пирамиды: $V=x y h$ (x, y - стороны, h - высота),

5. Создать базовый класс - фигура, и производные классы - круг, прямоугольник. Определить виртуальные функции площадь, периметр и вывод на печать.

6. Создать базовый класс - работник и производные классы - служащий с почасовой оплатой, служащий в штате. Определить функцию начисления зарплаты.

Тема 4. Шаблоны функций. Шаблоны классов. Паттерны проектирования.

Практические задания для самостоятельного решения

1. Создать класс студент, имеющий имя (указатель на строку), курс и идентификационный номер. Определить конструкторы, деструктор и функцию печати. Создать public-производный класс – студент-дипломник, имеющий тему диплома. Определить конструкторы по умолчанию и с разным числом параметров, деструкторы, функцию печати. Определить функции переназначения названия диплома и идентификационного номера.

2. Создать класс животное, имеющий классификацию (указатель на строку), число конечностей, число потомков. Определить конструкторы, деструктор и функцию печати. Создать public-производный класс - домашнее животное, имеющий кличку. Определить конструкторы по умолчанию и с разным числом параметров, деструкторы, функцию печати. Определить функции переназначения названия клички и числа потомков.

3. Создать класс машина, имеющий марку (указатель на строку), число цилиндров, мощность. Определить конструкторы, деструктор и функцию печати. Создать public-производный класс – грузовики, имеющий грузоподъемность кузова. Определить конструкторы по умолчанию и с разным числом параметров, деструкторы, функцию печати. Определить функции переназначения марки и грузоподъемности.

4. Создать класс карта, имеющая ранг и масть. Карту можно перевернуть и открыть. Создать класс – колода карт, содержащий карты. Создать два производных класса от колоды карт, в одном карты могут доставаться только по порядку, в другом - вытаскиваться произвольно.

5. Создайте класс точка, которая имеет координаты. Класс эллипсов, и класс окружностей. Определить иерархию типов. Определить функции печати, конструкторы, деструкторы, вычисление площади.

Критерии оценки:

0 баллов - ставится, если студент не выполнил лабораторную работу.

1 балл - ставится, если студент обнаруживает знание и понимание основных положений лабораторной работы, но при выполнении заданий допущены ошибки или задание выполнено на 50%; оформление работы выполнено недостаточно последовательно (отсутствуют цель/листинг/результаты/выводы).

2 балла - ставится, если студентом при выполнении заданий допущены неточности или задание выполнено на 70%; оформление работы выполнено с ошибками (отсутствуют цель/выводы).

3 балла - ставится, если студент полностью выполнил задание, правильно ответил на теоретические вопросы преподавателя, оформление работы выполнено последовательно и полно (присутствуют цели работы, задания, листинг программ, результаты и выводы).

Самостоятельная работа студента

Включает проработку конспектов лекций, обязательной и дополнительной учебной литературы в соответствии с планом занятия; выполнение заданий. Основной формой проверки СРС является устный фронтальный опрос на занятии или письменные ответы на вопросы для проверки знаний по теме.

Самостоятельная работа студента может быть выполнена печатным способом на листах формата А4. Работа должна быть защищена, то есть студент должен отвечать на вопросы преподавателя по тематике работы. По собственному желанию студент может подготовить презентацию к самостоятельной работе, в которой кратко дается описание работы с уместным использованием мультимедийных технологий (изображения, видео- или аудио-информация). В качестве информационных источников для подготовки работы студент может использовать рекомендованную литературу по данной дисциплине.

Темы заданий для самостоятельной работы студентов

СРС 1. Объектно-ориентированное программирование. Язык C++

СРС 2. Классы и объекты C++

СРС 3. Перегрузка операций в ООП. Наследование в ООП.

СРС 4. Шаблоны функций. Шаблоны классов. Паттерны проектирования.

Перечень теоретических вопросов:

1. Объектно-ориентированное проектирование: концептуальное описание задачи и построение объектной модели предметной области.
2. Объектно-ориентированное представление данных: понятие объекта; отличие понятия «объект» от понятия «данные» в других языках программирования.
3. Объектно-ориентированное представление данных: понятие объекта; поведение и внешний интерфейс объекта; сообщения и методы.
4. Объектно-ориентированное представление данных: понятие объекта; внутренняя структура, состояние, свойства и поведение объекта; принцип скрытия реализации.
5. Объектно-ориентированные системы программирования: основные свойства и преимущества. Объектно-ориентированное проектирование: принципы описания прикладных классов и программирования в объектно-ориентированной среде.
6. Технологии программирования. Модульное программирование. Структурное программирование. Их характеристика. Преимущества и недостатки.
7. Принципы объектного подхода. Инкапсуляция. Ее характеристики.
8. Принципы объектного подхода. Полиморфизм и его характеристики.
9. Принципы объектного подхода. Наследование и его характеристики.
10. Определите процедурную и объектную декомпозицию предметной области задачи. Чем они различаются? Назовите достоинства и недостатки этих способов декомпозиции.
11. Назовите основные принципы ООП и дайте им характеристику.
12. Что такое объект, и каким образом объекты соединяются в систему для решения задачи?
13. Чем характеризуется объект?
14. Определите понятие «класс». Чем классы отличаются от других типов данных?
15. Как связаны между собой объект предметной области, класс и программный объект?
16. Каким образом в программных объектах реализуются состояние, поведение и идентификация объектов предметной области? Назовите операции, которые могут быть выполнены над программными объектами.
17. Определите основные средства разработки классов. Почему они названы основными? Охарактеризуйте каждое из перечисленных средств и поясните в каких ситуациях их целесообразно использовать.

18. Какие дополнительные средства разработки классов появились в последние годы? Для чего они могут быть использованы?
19. Назовите основные этапы разработки программных систем с использованием ООП и расскажите о каждом из них.
20. Классификация полей класса по способу доступа к ним?
21. Особенности объектной модели в языке C++?
22. Указатель this и его использование при описании объектов в языке C.
23. Какие типы данных в языке C Вы знаете. Дайте характеристику каждому типу.
24. Охарактеризуйте формат описания класса в языке C++?
25. Опишите формат описания реализаций функций- членов класса
26. Классы и объекты языка C++. Соотношение этих понятий.
27. Приведите шаблон описания класса в языке C++ . Основные элементы описания. Интерфейсная часть описания и реализация.
28. Какие области видимости компонентов класса Вы знаете?. Дайте им характеристику и приведите примеры.
29. Константные поля, методы и объекты. Дайте им характеристику и приведите примеры.
30. Конструкторы. Назначение конструкторов. Объявление конструкторов. Привести примеры.
31. Виды конструкторов. Их назначение. Внешние отличия конструкторов. Способы вызова конструкторов при создании объектов.
32. Указатели и ссылки в языке C++. Их сравнительная характеристика. Привести примеры.
33. Возврат значение из функции C++ через параметры. Привести примеры.
34. Что такое static?
35. Разница между overloading (перегрузкой) и overriding (переопределением)?
36. Что такое указатель THIS?
37. Что такое namespace в ООП?
38. Что такое геттеры(getters)/сеттеры(setters)?
39. Что такое клонирование в ООП?

Критерии оценки:

0 баллов – самостоятельная работа не выполнена.

1 балл – демонстрирует, лишь поверхностный уровень выполнения работы, в содержании выполнения задания допущены принципиальные ошибки.

2 балла – ставится тогда, когда студент выполнил самостоятельную работу, но дает не точные ответы на заданные вопросы.

3 балла – ставится тогда, когда студент выполнил самостоятельную работу, показан высокий уровень освоения студентом учебного материала, содержание выполнения задания не содержит ошибок.

Контрольная работа

Контрольная работа предполагает выполнение письменной работы с обязательными практическими примерами по одной из тем.

Контрольная работа предполагает изучение теоретического материала по заданной теме, самостоятельное рассмотрение примеров, анализ различных источников, формулирование выводов. Работа выполняется печатным способом на листах формата А4, объемом 10-15 страниц. Работа должна быть защищена, то есть студент должен отвечать на вопросы преподавателя по тематике работы. Студент должен подготовить презентацию к работе, в которой кратко дается описание работы с уместным использованием мультимедийных технологий (изображения, видео- или аудио-информация).

Структура контрольной работы

Титульный лист

Оглавление

Введение

Содержание контрольной работы, состоящее из 2-3 глав, в достаточной мере описывающее предметную область с примерами.

Заключение

Список используемых источников

Приложения (при необходимости)

Тематика контрольных работ

1. Объектно-ориентированного программирования.
2. Понятие объектно-ориентированного программирования (ООП).
3. Понятие объекта и класса.
4. Основные принципы ООП.
5. Абстрагирование.
6. Инкапсуляция.
7. Агрегирование. Наследование.
8. Объектная модель программы.
9. Основные элементы классов.
10. Режимы доступа. Поля класса.
11. Методы класса. Тело метода. Вызов метода. Перегрузка методов.
12. Конструкторы класса.
13. Свойства класса. Автоматически реализуемые свойства.
14. Инициализация объектов класса. Индексаторы.
15. Статические поля и методы класса.
16. Переопределение операций класса.
17. Определение преобразования типов.
18. Отношение вложенности. Отношение наследования.
19. Описание производных классов. Конструкторы производного класса.
20. Добавление методов в производном классе.
21. Абстрактные классы.
22. Присвоение ссылок с учетом наследования.
23. Полиморфизм.
24. Наследование. Принцип подстановки и проблема «круг-эллипс», варианты ее решения. Множественное наследование и проблема ромба. Альтернативы множественному наследованию.
25. Объекты. Тип, идентичность, изменяемость, состояние объектов, объекты-значения и объекты-ссылки. тирования, их назначение, классификация и содержание.
26. Порождающие шаблоны проектирования: абстрактная фабрика, фабричный метод.
27. Порождающие шаблоны проектирования: строитель, прототип, объект-

одинок.

28. Структурные шаблоны проектирования: адаптер, декоратор, заместитель.

29. Структурные шаблоны проектирования: мост, фасад, приспособленец, компоновщик.

30. Поведенческие шаблоны проектирования: команда, цепочка ответственности, интерпретатор.

31. Поведенческие шаблоны проектирования: итератор, посредник, хранитель.

32. Поведенческие шаблоны проектирования: наблюдатель, стратегия, состояние.

Завершив контрольную работу, студент должен подготовить ее к защите, для чего он готовит выступление, презентацию, наглядную информацию для использования во время защиты.

В тексте выступления студент обосновывает актуальность избранной темы, проводит обзор научных работ по предложенной теме, показывает значимость исследования, дает краткий обзор глав работы и представляет полученные в процессе выполнения результаты.

Текст выступления должен быть максимально приближен к тексту контрольной работы, поэтому основу выступления составляют введение и заключение, которые используются в выступлении практически полностью. Также практически полностью используются выводы в конце каждой из глав.

Структура текста выступления при защите курсовой работы (курсового проекта) может быть примерно следующей:

Раздел выступления	Продолжительность (мин.)	Количество слайдов презентации
Введение	1-2	1-2
Краткое содержание работы (выводы по главам)	3-5	Не менее 1 на каждую главу
Заключение (основные выводы, дальнейшие перспективы разработки проблемы)	2-3	1-2

Основными принципами при составлении компьютерной презентации являются лаконичность, ясность, уместность, сдержанность, наглядность (подчеркивание

Критерии оценки:

0 баллов – контрольная работа не выполнена.

1-8 баллов – демонстрирует, лишь поверхностный уровень выполнения работы, в содержании выполнения задания допущены принципиальные ошибки, на заданные вопросы отвечает нечетко и неполно.

9-14 баллов – ставится тогда, когда студент выполнил контрольную работу, твердо знает материал, но дает не точные ответы на заданные вопросы, в содержании выполнения задания допущены не принципиальные ошибки.

15-16 баллов – ставится тогда, когда студент выполнил контрольную работу, показан высокий уровень освоения студентом учебного материала, содержание выполненного задания не содержит ошибок или допущены неточности, которые были устранены после замечаний, в работе присутствуют четкие и обоснованные выводы.

Аттестационная работа

Аттестационная работа проверяет знание студентов по изученному разделу. Проводится в форме тестирования.

Тестовые задания:

1. Объектно-ориентированное программирование характеризуется:
 - a) Наличием одной линейной программы
 - b) Разделением программы на модули
 - c) Все данные об объекте, его связи с другими объектами объединяются в одну структурную переменную
2. Метод это:
 - a) Функция, получая в качестве обязательного параметра указатель на объект
 - b) Структура, хранящая указатели `this`, `parent`, `super`
 - c) Структурная переменная, содержащая всю информацию о некотором физическом предмете или реализуемом в программе понятии
 - d) Определенный программистом абстрактный тип данных
3. Объект это:
 - a) Функция, получая в качестве обязательного параметра указатель на объект
 - b) Структура, хранящая указатели `this`, `parent`, `super`
 - c) Структурная переменная, содержащая всю информацию о некотором физическом предмете или реализуемом в программе понятии
 - d) Определенный программистом абстрактный тип данных
4. Класс это:
 - a) Структурная переменная, содержащая всю информацию о некотором физическом предмете или реализуемом в программе понятии
 - b) Определенный программистом абстрактный тип данных
 - c) Переменная, описанная абстрактным типом данных
5. Когда данные являются не глобальными, доступными всей программе, а локальными доступными только малой части программы:
 - a) При инкапсуляции
 - b) При полиморфизме
 - c) При объявлении метода
6. Когда функция определяется независимо в каждом производном классе и имеет в этих классах общее имя:
 - a) При инкапсуляции
 - b) При полиморфизме
 - c) При наследовании и инкапсуляции
7. Для выполнения операции присвоение одного объекта другому:
 - a) Достаточно чтобы в классе этих объектов был специальный конструктор - конструктор копирования
 - b) Классы этих объектов нужно описать одинаковым образом
 - c) Достаточно чтобы в классе этих объектов были конструкторы и деструкторы
 - d) Необходимо описать оба объекта одним именем класса и иметь в классе конструктор копирования
8. В каких случаях надо иметь в классе конструктор копирования:
 - a) Когда нужно передать элементы класса
 - b) Для выполнения операции присвоение одного элемента другому
 - c) Для создания дружественной функции
 - d) Для уничтожения объекта из памяти
9. В чем состоит задача конструктора копирования:
 - a) Четко определить порядок создания копии объекта
 - b) Размножить объект в указанном количестве копий
 - c) Копировать свойства других объектов

d) Создать this указатель

10. Как называются элементы класса которые относятся ко всем экземплярам объектов класса

- a) Статические
- b) Динамические
- c) Константные
- d) Защищенные

11. Какая процедура возвращает в кучу кусок памяти, зарезервированный за типизированным указателем? Напишите любую.
(маленькими строчными буквами, без скобок и аргументов)

Ваш ответ: _____

12. Какой принцип ООП 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) Полиморфизм
- b) Агрегация
- c) Композиция
- d) Инкапсуляция
- e) Персистентность

13. Выберите наиболее подходящее определение Класса.

- a) Тип, содержащий набор функций
- b) Тип, который отображает состояние некоторого объекта
- c) Тип, описывающий поведение некоторой сущности
- d) Тип, описывающий характеристики и поведение объекта

14. Наследование и композиция взаимоисключающие понятия. То есть при создании иерархии объектов используется либо наследование, либо композиция.

- a) Да
- b) Нет

15. К проявлению какого элемента ООП можно отнести следующую фразу: "Перемещая рукоятку коробки передач автомобиля, человек может не задумываться о самом механизме переключения. С точки зрения водителей все коробки передач работают одинаково, хотя их механизмы на самом деле могут отличаться".

- a) Наследование
- b) Инкапсуляция
- c) Полиморфизм

16. Как называется способность объекта скрывать свои данные и реализацию от других объектов системы?

- a) Полиморфизм
- b) Инкапсуляция
- c) Абстракция
- d) Наследование

17. Какая разница между идентичностью (identity) и равенством (equality) объектов в ООП?

- a) Идентичность означает, что у объектов есть общий неабстрактный предок, а равенство - любой общий предок
- b) Идентичность означает, что у объектов одинаковые поля, а равенство - что они содержат одинаковые данные
- c) Идентичность означает, что объекты являются экземплярами одного и того же класса, а равенство - что они содержат одинаковые данные

d) Идентичность означает, что две ссылки указывают на один и тот же объект, а равенство - что они содержат одинаковые данные

18. 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) Инкапсуляция
- b) Равенство
- c) Композиция
- d) Полиморфизм
- e) Агрегация

19. "Черный ящик" как способ повторного использования - это...

- a) Стиль повторного использования, основанный на наследовании классов
- b) Стиль повторного использования, основанный на композиции объектов.
- c) Название структурного паттерна проектирования
- d) Стиль повторного использования, основанный на сильной связанности объектов

20. Как в терминах ООП называется объект, который не может быть изменён после создания?

- a) Internal object
- b) Abstract object
- c) Immutable object
- d) Sealed object
- e) Все варианты
- f) Среди вариантов нет правильного ответа

Критерии оценки:

Процент выполненных тестовых заданий	Количество набранных баллов
91% - 100%	10
81% - 90%	9
71% - 80%	8
61% - 70%	7
51% - 60%	6
<50%	0