

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Игнатенко Виталий Иванович

Должность: Проректор по образовательной деятельности и молодежной политике

Дата подписания: 14.11.2023 05:28:38

Уникальный программный идентификатор:

a49ae343af5448d45d7e3e1e499659da8109ba78

**Министерство науки и высшего образования РФ**

**Федеральное государственное бюджетное образовательное учреждение  
высшего образования**

**«Запалярный государственный университет им. Н.М. Федоровского»**

**ЗГУ**

## **ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

**по дисциплине**

**«Объектно-ориентированное проектирование информационных систем»**

Факультет электроэнергетики, экономики и управления (ФЭЭ и У)

Направление подготовки: 09.04.03 «Прикладная информатика

Направленность (профиль): Информационные системы и технологии в бизнесе

Уровень образования: магистратура

Кафедра информационных систем и технологий

*наименование кафедры*

Разработчик ФОС:

доцент, к.т.н., доцент

(должность, степень, ученое звание)

(подпись)

А.А. Попкова

(ФИО)

Оценочные материалы по дисциплине рассмотрены и одобрены на заседании кафедры,  
ИСиТ

И.о.зав. кафедрой \_\_\_\_\_ И.С. Беляев

Фонд оценочных средств по дисциплине «*Объектно-ориентированное проектирование информационных систем*» для текущей/промежуточной аттестации разработан в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.04.03 «*Прикладная информатика*» на основе Рабочей программы дисциплины «*Объектно-ориентированное проектирование информационных систем*», утвержденной решением ученого совета № 12-3 от 28.04.2023, Положения о формировании Фонда оценочных средств по дисциплине (ФОС), Положения о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся ЗГУ, Положения о государственной итоговой аттестации (ГИА) выпускников по образовательным программам высшего образования в ЗГУ.

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

Таблица 1 – Компетенции и индикаторы их достижения

Код и наименование компетенции	Индикаторы достижения и планируемые результаты обучения по дисциплине
Универсальные	
«УК»	-
Общепрофессиональные	
«ОПК»	-
Профессиональные компетенции	
«ПК» ПК-3. Способен управлять процессом разработки программного обеспечения	ПК-3.2. Демонстрирует навыки планирования процесса разработки программного продукта

Таблица 2. – Паспорт фонда оценочных средств

Контролируемые разделы (темы) дисциплины	Формируемая компетенция	Наименование оценочного средства	Форма оценивания
Тема 1.1. Основные элементы проектирования ИС	ПК-3.2	Контрольные вопросы	Устно
Тема 1.2. Изучение основных возможностей CASE-средств по построению моделей бизнес-процессов и моделей данных	ПК-3.2	Контрольные вопросы,	Устно
Тема 1.3. Проектирование ИС.	ПК-3.2	Контрольные вопросы,	Устно

## 2. Перечень контрольно-оценочных средств (КОС)

Для определения качества освоения обучающимися учебного материала по дисциплине используются следующие контрольно-оценочные средства текущего контроля успеваемости, промежуточной аттестации обучающихся:

Таблица 3. Перечень контрольно-оценочных средств

	Наименование оценочного средства	Сроки выполнения	Шкала оценивания	Критерии оценивания
1.	<i>Текущий контроль качества</i>			
1	Контрольные вопросы	2 семестр	Достигнут/ не достигнут пороговый уровень освоения компетенции	Зачтено/ не зачтено
	<i>Промежуточная аттестация - экзамен</i>			
1	Тестовые задания	2 семестр	Освоил/ не освоил компетенцию	Зачтено/ не зачтено
2	Экзаменационные билеты	2 семестр	от 2 до 5 баллов	Критерии оценивания приведены ниже
	<p><b>Критерии оценивания тестовых заданий «зачтено», «не зачтено»:</b></p> <ul style="list-style-type: none"> <li>- «Зачтено» выставляется обучающемуся, если он показал достаточно прочные знания основных положений учебной дисциплины, умение самостоятельно решать конкретные практические задачи, предусмотренные рабочей программой, ориентироваться в рекомендованной справочной литературе, умеет правильно оценить полученные результаты.</li> <li>- «Не зачтено» выставляется обучающемуся, если при ответе выявились существенные пробелы в знаниях основных положений учебной дисциплины, неумение с помощью преподавателя получить правильное решение конкретной практической задачи из числа предусмотренных рабочей программой.</li> </ul>			
	<p><b>Критерии оценивания для экзамена по 4-балльной шкале:</b></p> <ul style="list-style-type: none"> <li>- оценки «отлично» заслуживает обучающийся, обнаруживший всесторонние, глубокие знания учебного материала и умение свободно выполнять задания, предусмотренные программой; изучивший основную литературу и знакомый с дополнительной литературой, рекомендованной программой обучения; безупречно отвечавший не только на вопросы билета, но и на дополнительные вопросы; проявивший творческие способности в использовании учебного материала;</li> <li>- оценки «хорошо» заслуживает обучающийся, обнаруживший полные знания учебного материала, успешно выполнивший предусмотренные программой задания, изучивший основную литературу, отвечавший на все вопросы билета;</li> <li>- оценки «удовлетворительно» заслуживает обучающийся, обнаруживший знания в объеме, необходимом для дальнейшей учёбы и работы по профессии,</li> </ul>			

<p>справившийся с выполнением заданий, знакомый с основной литературой, допустивший погрешности в ответе и при выполнении заданий, но обладающий достаточными знаниями для их устранения под руководством преподавателя;</p> <p>- оценка «неудовлетворительно» выставляется обучающемуся, обнаружившему пробелы в знаниях основного учебного материала, допустившему принципиальные ошибки в выполнении заданий, которые не позволят ему продолжить обучение без дополнительных занятий по соответствующей дисциплине.</p>
--

### **3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций в ходе освоения образовательной программы**

#### **3.1 Задания для текущего контроля успеваемости**

##### **Задания практических работ**

Практическая работа № 1. По теме: «Построение диаграмм бизнес- процессов и данных для заданной предметной области»

Практическая работа № 2. По теме: «Создание первого проекта»

Практическая работа № 3 По теме: «Интернет технологии проектирования ИС»

Практическая работа № 4 По теме: «Администрирование и оптимизация ИС»

#### **3.2 Задания для промежуточной аттестации**

##### **Контрольные вопросы к экзамену**

1. Что такое проект?
2. В чем состоит отличие проектной деятельности от операционной?
3. Какие формальные критерии можно использовать на предприятии для отнесения той или иной активности к проектной деятельности?
4. Назовите основные типы проектов в зависимости от степени уникальности результата и процесса.
5. Что такое «проектный треугольник»?
6. Опишите взаимосвязь основных элементов проекта.
7. Что из себя представляет и с какой целью разрабатывается «матрица компромиссов проекта»?
9. В чем состоит отличие понятий «проект», «программа», «портфель проектов»?
10. Назовите характерные особенности IT-проектов.
11. Кто относится к субъектам управления проектом?
12. Что является объектом управления в системе управления проектом?
13. Назовите и охарактеризуйте известные вам международные и национальные стандарты управления проектами.
14. Перечислите области знания в управлении проектами в соответствии с РМВоК.
15. Назовите и охарактеризуйте фазы жизненный цикл проекта;
16. Охарактеризуйте и представьте графически характер распределения затрат проекта во времени в соответствии с фазами жизненного цикла проекта.
17. Назовите группы процессов управления проектами в соответствии с РМВоК.
18. Чем отличается иерархическая структура работ от сетевого графика проекта?
19. Что такое критический путь в проекте?
20. Какие отношения определяют положение любой операции в графике проекта?
21. Что такое прямой анализ сетевого графика проекта?
22. Что такое обратный анализ сетевого графика проекта?
23. Какие преимущества для менеджера проекта дает знание резервов времени выполнения операций?
24. С какой целью при построении сетевых графиков используются отношения задержки операций?

25. Что такое «веха» и в чем состоит цель её использования?
26. Опишите функционал MS Project, используемый для календарного планирования.
27. Назовите и охарактеризуйте основные виды задач, используемые в MS Project.
28. Назовите и охарактеризуйте основные типы ресурсов, используемых в проектной деятельности.
29. Назовите причины превышения доступности ресурсов.
30. Как связаны календарное планирование ресурсов и приоритет проекта?
31. Какие операции задерживаются при выравнивании использования ресурсов?
32. Что такое «матрица ответственности», в чем состоит цель её использования?
33. Назовите основные особенности управления ресурсами IT-проектов.
34. Опишите функционал MS Project, используемый для создания ресурсов в проекте, назначения ресурсов, выравнивания ресурсов.
35. Что такое бюджет? В чем состоит отличие бюджета от сметы?
36. В чем состоит отличие прямых и косвенных затрат проекта?
37. Назовите типовые статьи затрат для IT-проекта.
38. Кто несет ответственность за формирование и исполнение бюджета проекта?
39. Что такое риск? Приведите известные вам классификации рисков.
40. Существуют ли риски, оказывающие положительное влияние на проект?
41. Назовите типичные риски IT-проектов.
42. В чем заключается качественный анализ рисков, какова цель его проведения?
43. В чем заключается количественный анализ рисков, какова цель его проведения?
44. В чем заключается необходимость применения метода освоенного объема?
45. Как определяются отклонения проекта по срокам его выполнения?
46. Как определяются отклонения проекта по стоимости?

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФГБОУ ВО «Заполярный государственный университет им. Н. М. Федоровского»  
КАФЕДРА ИНФОРМАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ**

**ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1**

К экзамену по дисциплине «Управление IT-проектами»

1. Назовите типичные риски IT-проектов.
2. В чем состоит отличие понятий «проект», «программа», «портфель проектов»?
3. На рисунке приведены данные о стадиях работ по созданию и внедрению IT-проекта на новом предприятии по переработке нефтепродуктов.

Стадия **H** должна выполняться специализированной организацией "IT-сервис". Стоимость работ \$8000. Однако организация " IT-сервис " может начать работы только на 6 недель позже запланированного в проекте «раннего» старта. Каждая неделя отсрочки окончания проекта стоит организаторам \$5000.

Преподаватель: \_\_\_\_\_

## Итоговый тест

### Вопросы на проверку компетенции ПК-3.

Способен управлять процессом разработки программного обеспечения

Оценочное средство (тестирования)	Контролируемые компетенции
<p>Что такое объектно-ориентированное программирование?</p> <p>A) Методология программирования, основанная на описании программных объектов и их взаимодействии                      B) Методология программирования, основанная на анализе и описании данных                      C) Методология программирования, основанная на проектировании баз данных                      D) Методология программирования, основанная на использовании SQL</p>	<b>ПК-3.2</b>
<p>Что такое класс в объектно-ориентированном программировании?</p> <p>A) Шаблон объекта, который определяет его свойства и методы                      B) Группа связанных таблиц в базе данных                      C) Метод для доступа к данным объекта                      D) Функция для преобразования типов данных</p>	<b>ПК-3.2</b>
<p>Какой принцип ООП гласит, что объект должен предоставлять только те свойства и методы, которые необходимы для работы с ним?</p> <p>A) Принцип единственной ответственности                      B) Принцип открытости/закрытости                      C) Принцип подстановки Барбары Лисков                      D) Принцип минимального интерфейса</p>	<b>ПК-3.2</b>
<p>Что такое наследование в объектно-ориентированном программировании?</p> <p>A) Механизм, который позволяет классам наследовать свойства и методы других классов                      B) Механизм, который позволяет классам иметь доступ к базе данных                      C) Механизм, который позволяет классам иметь доступ к файловой системе                      D) Механизм, который позволяет классам наследовать методы для доступа к данным</p>	<b>ПК-3.2</b>
<p>Что такое объектно-ориентированное программирование?</p> <p>a) Подход к программированию, основанный на объектах и классах.                      b) Подход к программированию, основанный на функциях и процедурах.                      c) Подход к программированию, основанный на типах данных.                      d) Подход к программированию, основанный на ассемблере.</p>	<b>ПК-3.2</b>
<p>Что такое класс в объектно-ориентированном программировании?</p> <p>a) Объект, созданный из класса.                      b) Набор функций, объединенных вместе.                      c) Шаблон, описывающий свойства и методы объектов.                      d) Ссылка на объект.</p>	<b>ПК-3.2</b>
<p>Что такое объект в объектно-ориентированном программировании?</p> <p>a) Экземпляр класса.                      b) Набор функций, объединенных вместе.                      c) Шаблон, описывающий свойства и методы объектов.                      d) Ссылка на класс.</p>	<b>ПК-3.2</b>
<p>Что такое наследование в объектно-ориентированном программировании?</p> <p>a) Возможность создавать новый класс на основе уже существующего.                      b) Возможность создавать новый объект на основе уже существующего.                      c) Возможность копировать свойства и методы одного объекта в другой.                      d) Возможность добавлять новые свойства и методы в уже существующий класс.</p>	<b>ПК-3.2</b>
<p>Что такое полиморфизм в объектно-ориентированном программировании?</p> <p>a) Возможность использовать один и тот же метод для разных классов.                      b) Возможность использовать разные методы для одного класса.                      c) Возможность добавлять новые свойства и методы в уже существующий класс.                      d) Возможность создавать новые классы на основе уже существующих.</p>	<b>ПК-3.2</b>
<p>Что такое инкапсуляция в объектно-ориентированном программировании?</p> <p>a) Способность объекта скрывать свои данные и методы от других объектов.                      b) Способность объекта обмениваться данными и методами с другими объектами.                      c) Способность объекта копировать свои данные и методы в другие объекты.                      d) Способность объекта наследовать свойства и методы другого объекта.</p>	<b>ПК-3.2</b>

Какой метод класса является конструктором? a) <b>init()</b> b) <b>new()</b> c) <b>del()</b> d) <b>str()</b>	<b>ПК-3.2</b>
Какой метод класса вызывается при удалении объекта? a) <b>init()</b> b) <b>new()</b> c) <b>del()</b> d) <b>str()</b>	<b>ПК-3.2</b>
Что такое абстрактный класс? a) Класс, который нельзя использовать для создания объектов. b) Класс, который содержит только абстрактные методы. c) Класс, который используется только для наследования. d) Класс, который не содержит свойств и методов.	<b>ПК-3.2</b>
Что такое интерфейс в объектно-ориентированном программировании? a) Способ взаимодействия объектов. b) Набор методов, которые должен реализовать класс. c) Набор свойств, которые должен реализовать класс. d) Способность класса наследовать свойства и методы другого класса	<b>ПК-3.2</b>
Какие модификаторы доступа используются в Python? a) public, private, protected b) public, private c) public, protected d) public	<b>ПК-3.2</b>
Что такое инкапсуляция данных? a) Способность объекта скрывать свои данные от других объектов. b) Способность объекта обмениваться данными с другими объектами. c) Способность объекта копировать свои данные в другие объекты. d) Способность объекта наследовать свойства и методы другого объекта.	<b>ПК-3.2</b>
Что такое наследование интерфейсов? a) Возможность класса реализовать несколько интерфейсов. b) Возможность класса наследовать свойства и методы другого класса. c) Возможность класса наследовать только интерфейсы. d) Возможность класса наследовать только абстрактные классы.	<b>ПК-3.2</b>
Что такое полиморфизм параметров? a) Возможность использовать разные параметры для одного метода. b) Возможность использовать один и тот же параметр для разных методов. c) Возможность использовать разные типы параметров для одного метода. d) Возможность использовать один и тот же тип параметра для разных методов.	<b>ПК-3.2</b>
Что такое агрегация в объектно-ориентированном программировании? a) Способность объекта использовать свойства и методы другого объекта. b) Включение одного объекта в другой объект. c) Создание нового объекта на основе уже существующего. d) Наследование свойств и методов от другого объекта.	<b>ПК-3.2</b>
Что такое перегрузка методов? a) Создание методов с одинаковым именем, но разным числом и типом параметров. b) Изменение свойств и методов унаследованного класса. c) Создание методов, которые возвращают разные типы данных. d) Изменение поведения метода при наследовании.	<b>ПК-3.2</b>
Какие операции можно выполнить с объектами в Python? a) Сложение, вычитание, умножение, деление b) Присваивание, сравнение, приведение типов. c) Сравнение, индексирование, итерация. d) Присваивание, сравнение, вызов методов.	<b>ПК-3.2</b>
Какие операции можно выполнить с объектами в Java? a) Сложение, вычитание, умножение, деление. b) Присваивание, сравнение, приведение типов. c) Сравнение, индексирование, итерация. d) Присваивание, сравнение, вызов методов.	<b>ПК-3.2</b>
Что такое множественное наследование? a) Возможность класса наследовать свойства и методы нескольких классов. b) Возможность создания нескольких объектов из одного класса. c) Возможность использовать несколько интерфейсов в одном классе. d) Возможность использовать несколько абстрактных классов в одном классе.	<b>ПК-3.2</b>

<p>Что такое компиляция? а) Процесс преобразования исходного кода в машинный код. б) Процесс исполнения машинного кода на компьютере. в) Процесс проверки исходного кода на ошибки. г) Процесс отладки программы.</p>	<b>ПК-3.2</b>
<p>Что такое интерпретация?  а) Процесс преобразования исходного кода в машинный код.  б) Процесс исполнения машинного кода на компьютере.  в) Процесс проверки исходного кода на ошибки.  г) Процесс выполнения исходного кода на компьютере</p>	<b>ПК-3.2</b>
<p>Что такое диаграмма классов в UML?  а) Диаграмма, показывающая связи между классами.  б) Диаграмма, показывающая процессы в системе.  в) Диаграмма, показывающая использование ресурсов в системе.  г) Диаграмма, показывающая последовательность действий в системе.</p>	<b>ПК-3.2</b>
<p>Что такое диаграмма последовательностей в UML?  а) Диаграмма, показывающая связи между классами.  б) Диаграмма, показывающая процессы в системе.  в) Диаграмма, показывающая использование ресурсов в системе.  г) Диаграмма, показывающая последовательность действий в системе.</p>	<b>ПК-3.2</b>
<p>Что такое SOLID?  а) Принципы объектно-ориентированного программирования.  б) Язык программирования.  в) Система управления базами данных.  г) Методология разработки программного обеспечения</p>	<b>ПК-3.2</b>
<p>Какое правило из SOLID описывает принцип единственной ответственности?  а) Single Responsibility Principle.  б) Open/Closed Principle.  в) Liskov Substitution Principle.  г) Interface Segregation Principle</p>	<b>ПК-3.2</b>
<p>Какое правило из SOLID описывает принцип открытости/закрытости?  а) Single Responsibility Principle.  б) Open/Closed Principle.  в) Liskov Substitution Principle.  г) Interface Segregation Principle.</p>	<b>ПК-3.2</b>
<p>Какое правило из SOLID описывает принцип подстановки Барбары Лисков?  а) Single Responsibility Principle.  б) Open/Closed Principle.  в) Liskov Substitution Principle.  г) Interface Segregation Principle.</p>	<b>ПК-3.2</b>
<p>Какое правило из SOLID описывает принцип разделения интерфейса?  а) Single Responsibility Principle.  б) Open/Closed Principle.  в) Liskov Substitution Principle.  г) Interface Segregation Principle.</p>	<b>ПК-3.2</b>
<p>Что такое паттерн проектирования?  а) Шаблон решения часто встречающихся проблем.  б) Программный код, выполняющий определенную функцию.  в) Система управления базами данных.  г) Фреймворк для разработки веб-приложений.</p>	<b>ПК-3.2</b>
<p>Какой паттерн проектирования используется для создания объектов без использования оператора new?  а) Фабричный метод.  б) Строитель.  в) Прототип.  г) Одиночка.</p>	<b>ПК-3.2</b>
<p>Какой паттерн проектирования используется для создания сложных объектов шаг за шагом? а) Фабричный метод.  б) Строитель.  в) Прототип.  г) Одиночка.</p>	<b>ПК-3.2</b>
<p>Какой паттерн проектирования используется для создания объектов на основе существующих объектов-прототипов?  а) Фабричный метод.  б) Строитель.</p>	<b>ПК-3.2</b>

<p>c) Прототип. d) Одиночка</p>	
<p>Какой паттерн проектирования используется для создания объекта, который может существовать только в единственном экземпляре? a) Фабричный метод. b) Строитель. c) Прототип. d) Одиночка.</p>	<b>ПК-3.2</b>
<p>Какой паттерн проектирования используется для создания объектов, которые должны изменять свое поведение в зависимости от состояния? a) Состояние. b) Стратегия. c) Шаблонный метод. d) Наблюдатель.</p>	<b>ПК-3.2</b>
<p>Какой паттерн проектирования используется для выбора из нескольких вариантов алгоритма, используемого в программе? a) Состояние. b) Стратегия. c) Шаблонный метод. d) Наблюдатель</p>	<b>ПК-3.2</b>
<p>Какой паттерн проектирования используется для определения базового алгоритма в суперклассе и переопределения некоторых его шагов в подклассах? a) Состояние. b) Стратегия. c) Шаблонный метод. d) Наблюдатель.</p>	<b>ПК-3.2</b>
<p>Какой паттерн проектирования используется для реализации механизма подписки на события? a) Состояние. b) Стратегия. c) Шаблонный метод. d) Наблюдатель.</p>	<b>ПК-3.2</b>
<p>Какой паттерн проектирования используется для представления иерархии типов, в которой каждый тип имеет один или несколько родительских типов? a) Прототип. b) Фабричный метод. c) Одиночка. d) Наследование.</p>	<b>ПК-3.2</b>
<p>Что такое абстрактный класс? a) Класс, который не может быть создан в объект. b) Класс, который может содержать как реализацию методов, так и их объявления. c) Класс, который может быть использован только для наследования. d) Класс, который не может содержать методы.</p>	<b>ПК-3.2</b>
<p>Что такое интерфейс? a) Класс, который может содержать как реализацию методов, так и их объявления. b) Класс, который может быть использован только для наследования. c) Класс, который не может содержать методы. d) Набор методов, которые должны быть реализованы классами, реализующими данный интерфейс.</p>	<b>ПК-3.2</b>
<p>Что такое агрегация в объектно-ориентированном проектировании? a) Отношение, когда объекты одного класса содержат ссылки на объекты другого класса. b) Отношение, когда объекты одного класса наследуют функциональность другого класса. c) Отношение, когда объекты одного класса создают объекты другого класса. d) Отношение, когда объекты одного класса управляют жизненным циклом объектов другого класса.</p>	<b>ПК-3.2</b>
<p>Что такое композиция в объектно-ориентированном проектировании? a) Отношение, когда объекты одного класса содержат ссылки на объекты другого класса. b) Отношение, когда объекты одного класса наследуют функциональность другого класса. c) Отношение, когда объекты одного класса создают объекты другого класса. d) Отношение, когда объекты одного класса управляют жизненным циклом объектов другого класса.</p>	<b>ПК-3.2</b>
<p>Что такое наследование в объектно-ориентированном проектировании? a) Отношение, когда объекты одного класса содержат ссылки на объекты другого класса. b) Отношение, когда объекты одного класса наследуют функциональность другого класса.</p>	<b>ПК-3.2</b>

<p>с) Отношение, когда объекты одного класса создают объекты другого класса.  д) Механизм, позволяющий создавать новый класс на основе уже существующего класса.</p>	
<p>Что такое полиморфизм в объектно-ориентированном программировании?  а) Способность объекта быть одновременно экземпляром нескольких классов.  б) Способность объекта изменять свою внутреннюю реализацию без изменения внешнего интерфейса.  с) Способность объекта принимать различные формы в зависимости от контекста использования.  д) Способность объекта работать с разными типами данных.</p>	<b>ПК-3.2</b>
<p>Что такое перегрузка методов в объектно-ориентированном программировании?  а) Возможность определения нескольких методов с одинаковым именем, но разными параметрами.  б) Возможность определения нескольких методов с одинаковым именем и параметрами.  с) Возможность определения нескольких методов с одинаковым именем, но разными типами возвращаемого значения.  д) Возможность определения нескольких методов с разными именами и параметрами.</p>	<b>ПК-3.2</b>
<p>Что такое перегрузка операторов в объектно-ориентированном программировании?  а) Возможность определения новых операторов для работы с объектами класса.  б) Возможность определения новых операторов для работы с базовыми типами данных.  с) Возможность переопределения стандартного поведения операторов.  д) Возможность создания новых методов, имитирующих работу операторов.</p>	<b>ПК-3.2</b>
<p>Что такое статический метод в объектно-ориентированном программировании?  а) Метод, который может быть вызван на уровне класса, без создания объекта.  б) Метод, который может быть вызван только на уровне объекта, после создания объекта.  с) Метод, который не может быть переопределен в классах-наследниках.  д) Метод, который может быть вызван только внутри класса, но не извне.</p>	<b>ПК-3.2</b>
<p>Что такое класс в C#?  а) Это объект, созданный на основе другого объекта.  б) Это набор данных и функций, объединенных в единую структуру.  с) Это метод, который может быть вызван на уровне класса, без создания объекта.  д) Это переменная, которая может хранить только одно значение.</p>	<b>ПК-3.2</b>
<p>Что такое объект в C#?  а) Это набор данных и функций, объединенных в единую структуру.  б) Это переменная, которая может хранить только одно значение.  с) Это метод, который может быть вызван на уровне класса, без создания объекта.  д) Это объект, созданный на основе другого объекта.</p>	<b>ПК-3.2</b>
<p>Что такое конструктор класса в C#?  а) Это метод, который может быть вызван на уровне класса, без создания объекта.  б) Это метод, который может быть вызван только на уровне объекта, после создания объекта.  с) Это метод, который вызывается автоматически при создании объекта.  д) Это метод, который вызывается автоматически при удалении объекта.</p>	<b>ПК-3.2</b>
<p>Какой модификатор доступа используется для членов класса, к которым можно обращаться только изнутри этого класса?  а) public  б) private  с) protected  д) internal</p>	<b>ПК-3.2</b>
<p>Какой модификатор доступа используется для членов класса, к которым можно обращаться из любого места в программе?  а) public  б) private  с) protected  д) internal</p>	<b>ПК-3.2</b>
<p>Какой модификатор доступа используется для членов класса, к которым можно обращаться из любого места в текущей сборке?  а) public  б) private  с) protected  д) internal</p>	<b>ПК-3.2</b>
<p>Что такое наследование классов в C#?  а) Это процесс создания объектов на основе других объектов.</p>	<b>ПК-3.2</b>

<p>b) Это механизм, позволяющий создавать новые классы на основе уже существующих.</p> <p>c) Это способность объекта принимать различные формы в зависимости от контекста использования.</p> <p>d) Это механизм, позволяющий изменять внутреннюю реализацию объектов без изменения внешнего интерфейса.</p>	
<p>Как в C# указать, что один класс наследует другой класс?</p> <p>a) С помощью ключевого слова "new".</p> <p>b) С помощью ключевого слова "using".</p> <p>c) С помощью ключевого слова "class".</p> <p>d) С помощью ключевого слова "inherits".</p>	<b>ПК-3.2</b>
<p>Что такое базовый класс в C#?</p> <p>a) Это класс, который служит основой для создания других классов.</p> <p>b) Это класс, который не имеет наследников.</p> <p>c) Это класс, который не имеет методов и свойств.</p> <p>d) Это класс, который не может быть создан в виде объекта.</p>	<b>ПК-3.2</b>
<p>Что такое виртуальный метод в C#?</p> <p>a) Это метод, который может быть вызван на уровне класса, без создания объекта.</p> <p>b) Это метод, который может быть вызван только на уровне объекта, после создания объекта.</p> <p>c) Это метод, который вызывается автоматически при создании объекта.</p> <p>d) Это метод, который может быть переопределен в производном классе.</p>	<b>ПК-3.2</b>
<p>Какой модификатор используется для объявления виртуального метода в C#?</p> <p>a) virtual</p> <p>b) override</p> <p>c) sealed</p> <p>d) abstract</p>	<b>ПК-3.2</b>
<p>Какой модификатор используется для переопределения виртуального метода в производном классе в C#?</p> <p>a) virtual</p> <p>b) override</p> <p>c) sealed</p> <p>d) abstract</p>	<b>ПК-3.2</b>
<p>Что такое абстрактный класс в C#?</p> <p>a) Это класс, который не может быть создан в виде объекта.</p> <p>b) Это класс, у которого есть хотя бы один абстрактный метод.</p> <p>c) Это класс, который не имеет методов и свойств.</p> <p>d) Это класс, который не может иметь наследников.</p>	<b>ПК-3.2</b>
<p>Какой модификатор используется для объявления абстрактного метода в C#?</p> <p>a) virtual</p> <p>b) override</p> <p>c) sealed</p> <p>d) abstract</p>	<b>ПК-3.2</b>
<p>Какой модификатор используется для переопределения абстрактного метода в производном классе в C#?</p> <p>a) virtual</p> <p>b) override</p> <p>c) sealed</p> <p>d) abstract</p>	<b>ПК-3.2</b>
<p>Что такое интерфейс в C#?</p> <p>a) Это класс, у которого есть хотя бы один абстрактный метод.</p> <p>b) Это класс, который не может быть создан в виде объекта.</p> <p>c) Это набор абстрактных методов и свойств, которые должны быть реализованы в производных классах.</p> <p>d) Это класс, который не может иметь наследников.</p>	<b>ПК-3.2</b>
<p>Как объявить интерфейс в C#?</p> <p>a) С помощью ключевого слова "interface".</p> <p>b) С помощью ключевого слова "class".</p> <p>c) С помощью ключевого слова "abstract".</p> <p>d) С помощью ключевого слова "virtual".</p>	<b>ПК-3.2</b>
<p>Какой модификатор доступа используется для методов и свойств, объявленных в интерфейсе в C#?</p> <p>a) public</p> <p>b) private</p> <p>c) protected</p> <p>d) internal</p>	<b>ПК-3.2</b>

Можно ли создать объект на основе интерфейса в C#? а) Да, можно. б) Нет, нельзя.	<b>ПК-3.2</b>
Как происходит реализация интерфейса в C#? а) С помощью наследования от класса интерфейса. б) С помощью реализации всех методов и свойств, объявленных в интерфейсе. в) С помощью наследования от класса, который уже реализует интерфейс. г) С помощью объявления новых методов и свойств в классе, которые соответствуют методам и свойствам интерфейса.	<b>ПК-3.2</b>
Что такое конструктор класса в C#? а) Это метод класса, который вызывается при создании объекта класса. б) Это метод класса, который вызывается при удалении объекта класса. в) Это метод класса, который вызывается при обновлении объекта класса. г) Это метод класса, который вызывается при сохранении объекта класса.	<b>ПК-3.2</b>
Какой модификатор доступа у конструктора класса по умолчанию в C#? а) public б) private в) protected г) internal	<b>ПК-3.2</b>
Может ли класс иметь более одного конструктора в C#? а) Да, может. б) Нет, не может.	<b>ПК-3.2</b>
Как называется конструктор, который принимает параметры в C#? а) Статический конструктор. б) Конструктор по умолчанию. в) Пользовательский конструктор. г) Конструктор базового класса.	<b>ПК-3.2</b>
Какой модификатор доступа у конструктора базового класса, вызываемого из конструктора производного класса в C#? а) public б) private в) protected г) internal	<b>ПК-3.2</b>

#### Ключи правильных ответов

Вопрос	Ответ
1.	A
2.	A
3.	D
4.	A
5.	A
6.	C
7.	A
8.	A
9.	A
10.	A
11.	A
12.	C
13.	B
14.	B
15.	B
16.	A
17.	A

18.	A
19.	C
20.	B
21.	A
22.	D
23.	D
24.	A
25.	A
26.	D
27.	A
28.	D
29.	A
30.	A
31.	B
32.	C
33.	D
34.	A
35.	A
36.	B
37.	C
38.	D
39.	A
40.	B
41.	C
42.	D
43.	D
44.	B
45.	D
46.	A
47.	C
48.	D
49.	C
50.	A
51.	A
52.	A
53.	B
54.	D
55.	C
56.	B
57.	A
58.	D
59.	B
60.	D
61.	A
62.	D
63.	A
64.	B
65.	D
66.	B
67.	C

68.	А
69.	А
70.	Нет,нельзя
71.	В
72.	А
73.	Public
74.	Да,может
75.	Пользовательский конструктор
76.	А
77.	Нет,нельзя

Объектно-ориентированное проектирование информационных систем (ООПИС) предоставляет возможность _____ и архитектуру программного обеспечения.	<b>ПК-3.2</b>
В ООПИС основной строительный блок представляет собой _____, который инкапсулирует состояние и поведение объекта.	<b>ПК-3.2</b>
Одним из принципов ООПИС является _____, позволяющее создавать иерархию классов, где более общие классы наследуются от более конкретных.	<b>ПК-3.2</b>
В ООПИС широко используется понятие _____, которое позволяет одному объекту получать доступ к данным и методам другого объекта.	<b>ПК-3.2</b>
ООПИС обеспечивает _____, позволяющую ограничить доступ к определенным членам класса и скрыть их от других объектов.	<b>ПК-3.2</b>
Одной из задач ООПИС является _____, позволяющая определить связи и зависимости между различными классами и объектами.	<b>ПК-3.2</b>
В ООПИС применяется понятие _____, которое позволяет разделить программу на отдельные компоненты, сосредоточенные на определенной функциональности.	<b>ПК-3.2</b>
ООПИС поддерживает принцип _____, который позволяет заменять объекты одного класса объектами другого класса с совместимым интерфейсом.	<b>ПК-3.2</b>
Одним из преимуществ ООПИС является возможность _____, что облегчает понимание и сопровождение программного кода.	<b>ПК-3.2</b>
ООПИС позволяет использовать _____, которые представляют собой шаблоны для создания объектов определенного типа.	<b>ПК-3.2</b>
В ООПИС активно используется понятие _____, которое позволяет организовать связи между объектами и определить их взаимодействие.	<b>ПК-3.2</b>
Одной из задач ООПИС является _____, которая позволяет определить и описать объекты, используемые в системе.	<b>ПК-3.2</b>
В ООПИС применяется понятие _____, которое позволяет разделить систему на независимые модули, связанные только через интерфейсы.	<b>ПК-3.2</b>
ООПИС предоставляет возможность _____, позволяющую создавать новые классы на основе существующих данных	<b>ПК-3.2</b>
Одной из основных концепций ООПИС является _____, которая позволяет ограничить доступ к данным и методам объекта только через его публичный интерфейс.	<b>ПК-3.2</b>

ООПИС обеспечивает возможность _____, позволяющую организовать связи между объектами и определить их взаимодействие. . В ООПИС применяется понятие _____, которое позволяет группировать связанные классы и объекты в единый контейнер	<b>ПК-3.2</b>
ООПИС поддерживает принцип _____, который позволяет объединять связанные классы и объекты в общую абстракцию.	<b>ПК-3.2</b>
Одной из задач ООПИС является _____, которая позволяет определить взаимодействие между объектами и последовательность выполнения операций.	<b>ПК-3.2</b>
В ООПИС применяется понятие _____, которое позволяет разделить систему на независимые компоненты с явно определенными интерфейсами взаимодействия.	<b>ПК-3.2</b>

#### Ключи на закрытый ФОС

Вопрос	Ответ
1	проектировать
2	Класс
3	Наследование
4	Объектная ссылка
5	Инкапсуляция
6	Моделирование
7	Пакеты
8	Полиморфизм
9	Переиспользование
10	Шаблоны
11	Ассоциация
12	Анализировать
13	Модули
14	Наследование
15	Взаимодействие
16	Контейнер
17	Абстракция
18	Моделирование
19	Компонент
20	интерфейс