

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Косенок Сергей Михайлович
Должность: ректор
Дата подписания: 20.06.2025 07:41:31
Уникальный программный ключ:
e3a68f3eaa1e62674b54f4998099d3d6b1ac7856

Тестовое задание для диагностического тестирования по дисциплине

Программирование на языках 4 GL, 6 семестр

Код, направление подготовки	09.03.01 Информатика и вычислительная техника
Направленность (профиль)	Искусственный интеллект и экспертные системы
Форма обучения	очная
Кафедра разработчик	Автоматизированных систем обработки информации и управления
Выпускающая кафедра	Автоматизированных систем обработки информации и управления

Тестовое задание для диагностического тестирования (6 семестр)

№	Проверяемая компетенция	Задание	Варианты ответов
1	ПК-12.2 ПК-12.3 ПК-13.1 ПК-13.2 ПК-13.3	Какой язык является объектно-ориентированным	1. Assembler 2. Prolog 3. C++ 4. C

2	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Объект - это ...	—
3	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Класс - это ...	—

4	ПК-12.2 ПК-12.3 ПК-13.1 ПК-13.2 ПК-13.3	Объектно-ориентированное программирование - это ...	1. методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определённого класса, а классы образуют иерархию наследования 2. методология программирования, основанная на представлении программы в виде совокупности моделей, каждый из которых является экземпляром определённого шаблона, а шаблоны образуют иерархию наследования 3. методология программирования, основанная на представлении программы в виде совокупности логических функций 4. методология программирования, основанная на представлении программы в виде модулей
---	---	---	---

5	ПК-12.2 ПК-12.3 ПК-13.1 ПК-13.2 ПК-13.3	В объектно-ориентированном программировании число является	1. полем 2. объектом 3. типом 4. переменной
---	---	--	--

6	ПК-12.2 ПК-12.3 ПК-13.1 ПК-13.2 ПК-13.3	Наследование - это ...	1. правила объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения 2. механизм объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения 3. концепция или механизм объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения 4. концепция объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения
---	---	---------------------------	---

7	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Полиморфизм - это ...	1. способность функции или предиката обрабатывать данные разных типов 2. способность функции обрабатывать данные разных подтипов 3. способность предиката обрабатывать данные разных типов 4. способность функции обрабатывать данные разных типов
8	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Расшифруйте аббревиатуру SOLID	1. single data, open–closed, Liskov substitution, interface segregation и dependency inversion 2. single responsibility, open–closed, Liskov substitution, interface segregation и dependency inversion 3. single responsibility, open–closed, Liskov substitution, interface segregation и dependency injection 4. single responsibility, open–connect, Liskov substitution, interface segregation и dependency inversion

9	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Максимальное количество деструкторов в классе	—
10	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Основные термины объектно- ориентированно го программирова ния	1. класс 2. граф 3. объект 4. сеть

11	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	... - это концепция объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения	—
12	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Абстракция - это ...	1. использование всех характеристик объекта, которые представлены в данной системе 2. использование только тех характеристик объекта, которые с достаточной точностью представляют его в данной системе 3. использование только эффективных характеристик объекта, которые имеются в данной системе 4. использование только не эффективных характеристик объекта, которые имеются в данной системе

13	ПК-12.2 ПК-12.3 ПК-13.1 ПК-13.2 ПК-13.3	... - это использование только тех характеристик объекта, которые с достаточной точностью представляют его в данной системе	—
14	ПК-12.2 ПК-12.3 ПК-13.1 ПК-13.2 ПК-13.3	Инкапсуляция - это ...	1. правила или утверждения, позволяющий разграничивать доступ к различным частям компонента 2. механизм сокрытия, позволяющий разграничивать доступ к различным частям компонента 3. механизм переадресации, позволяющий осуществлять доступ к различным компонентам 4. правила сокрытия, позволяющий разграничивать доступ к различным частям компонента

15	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Сопоставьте ключевые слова в C#	1. class ↔ abstract 2. method ↔ operator 3. static ↔ virtual
16	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Расставьте фрагменты кода на C# в правильном порядке	1. } 2. interface 3. { 4. private 5. IAnimal 6. string Name{get;set;}

17	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Расставьте фрагменты кода на С# в правильном порядке	1. Car 2. } 3. public Car() 4. { 5. class 6. { 7. public 8. }
18	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Расставьте фрагменты кода на С# в правильном порядке	1. { 2. static 3. } 4. public 5. void 6. Run()

19	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	Расставьте фрагменты кода на С# в правильном порядке	1. {} 2. private void 3. Cat 4. } 5. internal 6. Jump() 7. class 8. {
20	ПК-12.2 ПК-13.1 ПК-13.3	ПК-12.3 ПК-13.2	К принципам SOLID относиться	1. single data, open–closed 2. Liskov substitution, interface segregation и dependency inversion 3. single responsibility, open–closed 4. Liskov substitution, interface segregation и dependency injection