

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Косенок Сергей Михайлович

Должность: ректор

Дата подписания: 19.06.2024 07:24:06

Уникальный программный ключ:

e3a68f3eaa1e62674b54f4998099d3d6b1dct856

## Тестовое задание для диагностического тестирования по дисциплине:

Объектно-ориентированное программирование

Код, направление подготовки	09.03.02 Информационные системы и технологии
Направленность (профиль)	Безопасность информационных систем и технологий
Форма обучения	очная
Кафедра разработчик	Информатики и вычислительной техники
Выпускающая кафедра	Информатики и вычислительной техники

№	Проверяемая компетенция	Задание	Варианты ответов	Тип сложности вопроса
1	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Класс - это ...		Низкий
2	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Объект - это ...		Низкий

3	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Какой язык является объектно- ориентированным	1. Assembler 2. Prolog 3. C 4. C++	Низкий
---	--	---	---	--------

4	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Объектно-ориентированное программирование - это ...	<p>1. методология программирования, основанная на представлении программы в виде совокупности логических функций</p> <p>2. методология программирования, основанная на представлении программы в виде совокупности моделей, каждый из которых является экземпляром определённого шаблона, а шаблоны образуют иерархию наследования</p> <p>3. методология программирования, основанная на представлении программы в виде модулей</p> <p>4. методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определённого класса, а классы образуют иерархию наследования</p>	Низкий
---	--	--	--	--------

5	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	В объектно-ориентированном программировании число является	1. объектом 2. типом 3. переменной 4. полем	Низкий
6	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Максимальное количество деструкторов в классе	1. 1 шт. 2. 2 шт. 3. 4 шт. 4. 0 шт.	Средний
7	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Сопоставьте ключевые слова в C#	1. class <=> virtual 2. method <=> operator 3. static <=> abstract	Средний
8	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Основные термины объектно-ориентированного программирования	1. класс 2. граф 3. сеть 4. объект	Средний

9	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	... - это концепция объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения		Средний
10	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	... - это использование только тех характеристик объекта, которые с достаточной точностью представляют его в данной системе		Средний

11	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Абстракция - это ...	<ol style="list-style-type: none"> <li>1. использование всех характеристик объекта, которые представлены в данной системе</li> <li>2. использование только эффективных характеристик объекта, которые имеются в данной системе</li> <li>3. использование только не эффективных характеристик объекта, которые имеются в данной системе</li> <li>4. использование только тех характеристик объекта, которые с достаточной точностью представляют его в данной системе</li> </ol>	Средний
----	--	----------------------	---	---------

12	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Инкапсуляция - это ...	<ol style="list-style-type: none"> <li>1. механизм сокрытия, позволяющий разграничивать доступ к различным частям компонента</li> <li>2. механизм переадресации, позволяющий осуществлять доступ к различным компонентам</li> <li>3. правила или утверждения, позволяющий разграничивать доступ к различным частям компонента</li> <li>4. правила сокрытия, позволяющий разграничивать доступ к различным частям компонента</li> </ol>	Средний
----	--	---------------------------	--	---------

13	<p>ПК-2.1  ПК- 2.2  ПК-2.3  ПК-1.1  ПК-14.1  ПК-14.2  ПК-14.3</p>	<p>Наследование - это  ...</p>	<p>1. правила объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения</p> <p>2. механизм объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения</p> <p>3. концепция объектно-ориентированного программирования, согласно которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения</p> <p>4. 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</p>	<p>Средний</p>
----	---	------------------------------------	---	----------------



			<p>которой абстрактный тип данных может наследовать данные и функциональность некоторого существующего типа, способствуя повторному использованию компонентов программного обеспечения</p>	
--	--	--	--	--

14	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Полиморфизм - это ...	<ol style="list-style-type: none"><li>1. способность функции обрабатывать данные разных типов</li><li>2. способность функции или предиката обрабатывать данные разных типов</li><li>3. способность функции обрабатывать данные разных подтипов</li><li>4. способность предиката обрабатывать данные разных типов</li></ol>	Средний
----	--	--------------------------	--	---------

15	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Расшифруйте аббревиатуру <b>SOLID</b>	<ol style="list-style-type: none"> <li>1. single data, open–closed, Liskov substitution, interface segregation и dependency inversion</li> <li>2. single responsibility, open–closed, Liskov substitution, interface segregation и dependency inversion</li> <li>3. single responsibility, open–closed, Liskov substitution, interface segregation и dependency injection</li> <li>4. single responsibility, open–connect, Liskov substitution, interface segregation и dependency inversion</li> </ol>	Средний
----	--	---	---	---------

16	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	К принципам SOLID относятся	1. Liskov substitution, interface segregation и dependency inversion 2. single responsibility, open– closed 3. Liskov substitution, interface segregation и dependency injection 4. single data, open–closed	Высокий
17	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Расставьте фрагменты кода на С# в правильном порядке	1. Car 2. class 3. public 4. } 5. { 6. public Car() 7. } 8. {	Высокий

18	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Расставьте фрагменты кода на C# в правильном порядке	1. public 2. { 3. void 4. Run() 5. static 6. }	Высокий
19	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Расставьте фрагменты кода на C# в правильном порядке	1. { 2. Cat 3. } 4. { } 5. private void 6. internal 7. Jump() 8. class	Высокий

20	ПК-2.1 ПК- 2.2 ПК-2.3 ПК-1.1 ПК-14.1 ПК-14.2 ПК-14.3	Расставьте фрагменты кода на C# в правильном порядке	1. } 2. string Name{get;set;} 3. { 4. private 5. IAnimal 6. interface	Высокий
----	--	---	--	---------