

Тестовое задание для диагностического тестирования по дисциплине:

Проектная деятельность

Код, направление подготовки	09.03.04 ПРОГРАММНАЯ ИНЖЕНЕРИЯ
Направленность (профиль)	Программное обеспечение компьютерных систем
Форма обучения	Заочная
Кафедра-разработчик	Автоматики и компьютерных систем
Выпускающая кафедра	Автоматики и компьютерных систем

КУРС 1

Проверяемая компетенция	Задание	Варианты ответов	Тип сложности вопроса	Кол-во баллов
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	1. Что такое GitHub?	<ol style="list-style-type: none"> 1. Программа для работы с Git 2. Драйвер для Git 3. Веб-сервис для хостинга IT-проектов и их совместной разработки, основанный на Git 4. UI для работы с локальной версией Git 	Низкий	2
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	2. Что такое репозиторий Git?	<ol style="list-style-type: none"> 1. Любая директория/папка в моей ОС 2. Любая папка, находящаяся внутри Git 3. Репозиторий Git представляет собой каталог файловой системы, в котором находятся файлы конфигурации репозитория, файлы журналов, хранящие операции, выполняемые над репозиторием, индекс, описывающий расположение файлов, и хранилище, содержащее собственно файлы 4. Папка .git/ и все входящие в нее 	Низкий	2
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	3. Что делает команда git status?	<ol style="list-style-type: none"> 1. Показывает состояние проекта 2. Показывает имя и email пользователя 3. Показывает место, занимаемое репозиторием 4. Такой команды нет, есть только команда git show 	Низкий	2
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	4. Что делает команда git add?	<ol style="list-style-type: none"> 1. Создает файл с указанным именем и сразу добавляет его в Git 2. Добавляет локальный файл в удаленный репозиторий так, чтобы другие участники проекта могли его видеть 3. Это алиас/синоним команды git commit 4. Начинает отслеживать указанный файл или файлы 	Низкий	2
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	5. Что означает статус файла untracked в выводе команды git status?	<ol style="list-style-type: none"> 1. Что система Git не отслеживает этот файл 2. Что файл был удален из Git 3. Что файл находится вне репозитория Git 4. Что файл добавлен в .gitignore 	Низкий	2

УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	6. Что означает статус файла new в выводе команды git status?	<ol style="list-style-type: none"> 1. Что файл только что был создан и еще не отслеживается системой Git 2. Что файл только начал отслеживаться Git и пока не имеет истории 3. Что файл удаляли из Git и потом восстановили командой git return 4. Такого статуса нет, есть только статус deleted file 	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	7. Чем отличается master и origin master?	<ol style="list-style-type: none"> 1. Это просто два разных названия одной ветки 2. master принадлежит локальному репозиторию, а origin master - удаленному 3. Это две разные ветки локального репозитория 4. Ветки origin master не существует 	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	8. Как узнать, кто автор строки в файле, используя систему Git?	<ol style="list-style-type: none"> 1. Команда git show --author 2. Команда git commit --author 3. Команда git blame 4. Команда git status 	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	9. Что делает команда git stash?	<ol style="list-style-type: none"> 1. Отменяет все изменения 2. Сохраняет все изменения в буфер 3. Удаляет все измененные файлы 4. Такой команды нет 	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	10. Как сделать ветку с названием my_branch	<ol style="list-style-type: none"> 1. Команда: git branch my_branch 2. Команда: git create branch my_branch 3. Команда: git commit origin my_branch 4. Команда: git checkout my_branch 	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	11. Что сделает команда "git branch" без какого-либо параметра?	<ol style="list-style-type: none"> 1. Переключится на последнюю используемую ветку 2. Выведет ошибку 3. Выведет список локальных веток 4. Выведет список удаленных (remote) веток 	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	12. Почему бывают конфликты при слиянии веток?	<ol style="list-style-type: none"> 1. Потому что ветки были созданы в разное время 2. Потому что ветки были созданы от разных коммитов 3. Потому что в обеих ветках есть изменения одних и тех же строк 4. Это устаревшая проблема, ее нет с версии Git 1.2 	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	13. Для чего надо добавлять файлы в .gitignore?	<ol style="list-style-type: none"> 1. Чтобы Git удалял их историю, храня только последнюю версию 2. Чтобы Git при работе с ними переспрашивал "Are you sure you want to interact with this file?" 3. Чтобы Git не замечал их и любые команды Git не могли их заафектить 4. Файл .gitignore не несет никакой смысловой нагрузки, так что этого не надо делать 	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	14. Как добавить новую директорию в Git?	<ol style="list-style-type: none"> 1. Добавить каждый файл из этой директории в Git 2. Добавить хотя бы один файл из этой директории в Git 3. Никак, Git работает только с файлами, т.к. для директорий не бывает изменений и истории 4. Команда: git add -d 	Средний	5
УК-1, УК-2.1, УК-2.2, УК-	15. Чем директория с репозиторием	<ol style="list-style-type: none"> 1. Ничем, такая же директория 	Средний	5

2.5, ПК-4.1, ПК-8.1, ПК-10.1	отличается от любой другой?	2. Правами доступа - у директории-репозитория права доступа только того пользователя, который его "склонил" (git clone) 3. Наличием папки .git с настройками репозитория 4. Эта директория прописана в реестре ОС		
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	16. Как отменить действие команды "git add" на файл?	1. Команда git abort 2. Команда git stash 3. Команда git not-add 4. Команда git reset	Высокий	8
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	17. Как привести измененный файл в начальное состояние (до изменения)?	5. Команда git abort path/to/file 6. Команда git checkout path/to/file 7. Команда git pull path/to/file 8. Команда git commit path/to/file	Высокий	8
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	18. Как сделать коммит?	1. Всего лишь набрать команду git commit в любой момент времени 2. Сделать изменения в файлах и перечислить их после git commit. Например так: git commit a.file, b.file 3. Сделать изменения, собрать эти изменения командой "git add" или "git commit -a" и указать коммит-сообщение после ключа "-m" 4. Нельзя сделать коммит, ведь такого понятия не существует	Высокий	8
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	19. Как узнать, какие изменения мы сделали локально относительно последнего состояния нашего репозитория?	1. Команда git show 2. Команда git diff 3. Команда git izmeneniya 4. Команда git commit	Высокий	8
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	20. Что делает команда git stash?	1. Отменяет все изменения 2. Сохраняет все изменения в буфер 3. Удаляет все измененные файлы 4. Такой команды нет	Высокий	8

КУРС 2

Проверяемая компетенция	Задание	Варианты ответов	Тип сложности вопроса	Кол-во баллов
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	1. С помощью какой функции PHP можно прочитать файл?	1. file_get_contents 2. print_r 3. array_merge 4. require	Низкий	2
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	2. Что будет в переменной \$result после выполнения кода \$result = require 1.php, если в 1.php написан код <?php return 7 % 4;?	1. 2 2. 4 3. 7 4. 3	Низкий	2
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	3. Что будет в переменной \$x после выполнения кода \$x = 2; \$x -= 2;?	1. 2 2. 0 3. -2 4. 4	Низкий	2

УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	4. Как получить данные POST-запроса?	1. через функцию <code>getPostData()</code> 2. через константу <code>POST</code> 3. через переменную <code>\$_POST</code> 4. через функцию <code>request()</code>	Низкий	2
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	5. Что будет в переменной <code>\$result</code> после выполнения кода <code>\$i = 5; \$result = \$i++;</code> ?	1. 4 2. 7 3. 5 4. 6	Низкий	2
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	6. Как нужно создавать новый массив?	1. <code>\$a = new Array();</code> 2. <code>\$a = null;</code> 3. <code>\$a = [];</code> 4. <code>\$a = array();</code>	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	7. С помощью какой функции можно удалить Cookie?	1. <code>getcookie</code> 2. <code>deletcookie</code> 3. <code>setcookie</code> 4. <code>readcookie</code>	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	8. Какая из этих конструкций не относится к циклам?	1. <code>isset</code> 2. <code>foreach</code> 3. <code>for</code> 4. <code>While</code>	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	9. Как присвоить переменной значение?	5. <code>\$x === 5</code> 6. <code>\$x => 5</code> 7. <code>\$x == 5</code> 8. <code>\$x = 5</code>	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	10. С помощью какой функции можно выделить слово из предложения?	1. <code>strpos()</code> 2. <code>substr()</code> 3. <code>substr()</code> 4. <code>strchr()</code>	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	11. Когда вызывается деструктор?	1. при создании объекта 2. при создании класса 3. при удалении класса 4. при удалении объекта	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	12. Для чего нужен оператор <code>self</code> ?	1. для получения названия объекта 2. для доступа к статическим свойствам и методам 3. для того же, что и <code>\$this</code> 4. реализации методов класса	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	13. Интерфейсы нужны для того, чтобы	1. классы правильно наследовались 2. определять реализацию методов 3. определять какие методы будут в классе, реализующем интерфейс 4. для инкапсуляции	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1,	14. Абстрактный класс в PHP это	1. класс без свойств 2. класс без методов	Средний	5

ПК-8.1, ПК-10.1		3. класс, экземпляры которого нельзя создать 4. нету такого понятия		
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	15. Константа класса определяется при помощи ключевого слова	1. constant 2. define 3. static 4. const	Средний	5
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	16. final class означает, что	1. он не может быть унаследован 2. может быть создан лишь 1 его экземпляр 3. он не является абстрактным 4. не содержит констант	Высокий	8
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	17. Для чего предназначено ключевое слово throw?	1. для генерации ошибки 2. для прекращения работы скрипта 3. для бросания исключения 4. для удаления класса	Высокий	8
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	18. Переменная \$this нужна для	1. все того же что и остальные переменные 2. обращения к свойствам и методам объекта 3. обращения к классу 4. обращения к объекту внутри класса	Высокий	8
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	19. Что означает protected	1. свойства и методы доступны в экземплярах самого класса и в экземплярах наследников 2. свойства и методы доступны только в экземплярах наследника 3. обратное к public 4. то же самое что private	Высокий	8
УК-1, УК-2.1, УК-2.2, УК-2.5, ПК-4.1, ПК-8.1, ПК-10.1	20. Какое ключевое слово позволяет обращаться к родительскому классу?	1. protected 2. parent 3. self 4. this	Высокий	8