Тест 1

1. Что из нижеперечисленного относится к **преимуществам** водопадной модели?

*Модель простая, четкая и понятная. Её легко внедрить в рабочий процесс благодаря простым фазам, которые идут строго по порядку*

1. Что из нижеперечисленного относится к **недостаткам** водопадной модели?

*Фаза тестирования наступает в конце проекта, поэтому проблемы и ошибки могут быть выявлены не сразу – а устранять их будет дорого и сложно*

1. Что такое "итеративная модель"?

*это процесс создания программного обеспечения, который осуществляется небольшими этапами*

1. Что такое "спиральная модель"?

*Это модель процесса разработки программного обеспечения с учетом рисков*

1. Что такое "экстремальное программирование"?

*гибкая методология разработки программного обеспечения, в которой взяли лучшие практики гибкой разработки и усилил до максимума*

1. Что из нижеперечисленного относится к **преимуществам** итеративной модели?

*Быстрый выпуск минимально ценного продукта (MVP) и возможность вывести продукт на рынок и начать эксплуатацию гораздо раньше.*

1. Что из ниже перечисленного относится к **недостаткам** итеративной модели?

*Нет фиксированного бюджета и сроков, а также нужна сильная вовлеченность Заказчика в процесс*

1. Что из нижеперечисленного относится к **преимуществам** спиральной модели?

*Мониторинг рисков является одной из главных особенностей, делающих данную модель особенно привлекательной в том случае, если предстоит управление большим, сложным и дорогостоящим проектом*

1. Что такое "бережливое производство"?

*гибкая методология разработки программного обеспечения, в которой взяли лучшие практики гибкой разработки и усилил до максимума //Неверно*

1. Что такое "Behavior Driven Development"?

*подход к разработке программного обеспечения, сосредоточенный на поведении системы в целом*

1. Что такое "Test Driven Development"?

*подход к разработке программного обеспечения, при котором сперва пишутся тесты для функциональности*

1. Что такое "Feature Driven Development"?

*подход к разработке программного обеспечения, направленный на создание небольших функций или функциональных блоков*

Тест 2

1. Что из ниже перечисленного относятся к недостаткам Agile?

*У проекта нет чёткого плана и структуры*

2. Что такое "daily"?

*ежедневный отчет о проделанной участником работе и плане на следующий день*

3. Какое из ниже перечисленных утверждений сравнения Scrum и Kanban НЕВЕРНО?

*Scrum позволяет начать с того, что есть сейчас: взять принципы, уже присутствующие в компании, и постепенно их улучшать, а Kanban требует более тяжелое внедрение*

4. Какого типа анализа кода нет?

*регрессионный*

5. Что такое "CD"?

*автоматизация развертывания приложений в различные окружения*

 6. Что такое "спринт"?

*период, в течение которого команда создает и совершенствует отдельную часть продукта*

7. Что такое "Инструментарий программирования"?

*программные средства, которые помогают разработчикам создавать,   тестировать и поддерживать программное обеспечение*

     8. Для каких целей используется CD-конвейер?  
               *все вышеперечисленное*

     9. Что из ниже перечисленного относятся к преимуществу Agile?.  
     *Гибкость и открытость к любым изменениям*

 10. Что такое “доска”?  
 *пространство, отражающие стадии работы над продуктом*

11. Что такое "IDE"?  
 *инструмент, который объединяет в себе редактор кода, компилятор, отладчик и другие необходимые функции  
12. Что такое "CI"?   
 методология разработки и набор практик, при которых в код вносятся небольшие изменения с частыми коммитами*

13. Какой роли НЕТ в Kanban? *специально нанятый сотрудник, ведущий команду к достижению цели*

Тест 3

1. Какое из этих уровней будет выступать "посредником" между другими?

*уровень предметной области*

2. За что отвечает "представление"?

*используется для отображения данных пользователю*

3. Зачем нужен балансировщик нагрузки?

*для равномерного распределения нагрузки на серверы*

4. Какое из описаний соответствует архитектуре CQRS?

*вместо того, чтобы использовать единый интерфейс для обеих операций, паттерн предлагает использовать различные модели данных для команд и запросов*

5.   Какое из этих определений относится к определению поведенческих паттернов?

*паттерны рассматривают вопросы о связях между объектами и распределением обязанностей между ними. Для этого могут использоваться механизмы, основанные как на наследовании, так и на композиции.*

    6. Какое из этих определений описывает концепцию DAL?

*обеспечивает возможность смены хранилища данных без изменения бизнес-логики проекта*

    7. За что отвечает "контроллер"?

*является связующем звеном между другими элементами архитектуры*

    8. Какое из описаний соответствует событийной архитектуре?

*состоит из очень независимых и высоко-специализированных компонентов обработки событий, которые асинхронно получают и обрабатывают события*

    9. Какое из этих определений относится к определению структурных паттернов?

*паттерны предназначены для выстраивания правильной иерархии классов (упрощение создания новых классов, расширение существующих) и правильного взаимодействия между классами (уменьшение взаимосвязанности и т.п.).*

*10. Что такое "уровень"?  
 группа модулей, предоставляющих взаимосвязанный набор сервисов*

11. Какого типа клиента НЕ существует?

*худого*

12.Какое из описаний соответствует микросервисной архитектуре?

*сервисы одной системы могут быть написаны на различных языках программирования и общаться друг с другом, используя различные протоколы*

13. За что отвечает "модель"?

*содержит бизнес-логику приложения*

1. Какое из этих описаний относится к одноуровневому шаблону клиент-сервер?

*такая, где все прикладные программы рассредоточены по рабочим станциям, которые обращаются к общему серверу баз данных или к общему файловому серверу*

1. Какое из этих определений относится к определению порождающих паттернов?

*паттерны проектирования предназначены для создания объектов, позволяя системе оставаться независимой как от самого процесса порождения, так и от типов порождаемых объектов.*

1. За что отвечает "представление"?

*используется для отображения данных пользователю*

1. Какое из этих определений относится к определению поведенческих паттернов?

*(сам ответить не успел, гпт говорит этот ответ) паттерны рассматривают вопросы о связях между объектами и распределением обязанностей между ними. Для этого могут использоваться механизмы, основанные как на наследовании, так и на композиции.*

Тест 4

1. Что такое десериализация?

*это процесс восстановления состояния объекта с целью последующего использования.*

1. Чем характеризуется пользовательский подход?  
   *Подразумевает применение различных манипуляций с данными объекта класса перед сериализацией данных, в процессе сериализации, и в процессе или по окончании процесса десериализации данных*
2. Чем характеризуется бинарная сериализация  
   *компактность записи данных и сложность «ручного чтения» сериалиованных данных*
3. Что такое XSD?  
   *язык описания структуры XML документа*
4. Что такое JSON?  
   *текстовый формат обмена данными, основанный на JavaScript*
5. Какой из этих недостатков не относится к сериализации?  
   *построение неэффективных запросов*
6. Зачем нужен объект SerializationInfo?  
   *содержит* все *данные, необходимые для сериализации или десериализации объекта.*
7. Чем характеризуется SOAP сериализация  
   *имеет больше количество дополнительной информации, описывабщих структуру сериализуемого объекта*
8. Что такое тег?  
   *текст (название тега), обернутый в символы ’<’ ’>’*
9. Чем характеризуется базовый подход?

*Допускаются настройки для класса, указывающие, что он может быть сериализован и настройки для полей, что в процессе сериализации они не участвуют и информацию с них не сохранять*

1. Какая из этих часте*й НЕ обязательная в составе SOAP?*

*Заголовок*

1. Что такое JSON-объект?

*неупорядоченное множество пар «ключ:значение», заключаенное в ’{’ ’}’*

1. Что такое XML?

*расширяемый язык разметки, используемый для хранения и передачи данных*

1. *Что такое сериализация?  
     
   это процесс преобразования объекта в поток байтов для сохранения или передачи в память, базу данных или файл.*

*Тест 5*

1. Зачем нужен пул потоков?

*для уменьшения временных затрат на создание потока*

*(Пул потоков позволяет повторно использовать уже созданные потоки, избегая накладных расходов на их создание и уничтожение.)*

2. Какая характеристика относится к основному потоку?

*основной поток отвечает за функционирование интерфейса пользователя в десктопном приложении*

*(Главный (основной) поток обычно управляет UI в приложениях WPF/WinForms. Если он блокируется, интерфейс "зависает".)*

3. Какого из этих этапов нет в процессе проектирования распараллеленной программы?

*Распараллеливание*

*(Основные этапы: Декомпозиция (разделение задачи), Коммуникация (организация обмена данными), Укрупнение (оптимизация под архитектуру). "Распараллеливание" — это общая цель, а не этап.)*

4. Какой из механизмов синхронизации потоков может блокировать одновременный запуск нескольких приложений?

*Mutex*

*(Именованный Mutex может использоваться для межпроцессной синхронизации, блокируя выполнение кода в разных экземплярах приложения. Остальные (lock, Monitor, AutoResetEvent, Semaphore) работают только в пределах одного процесса.)*

5. Какая из потокобезопасных коллекций позволяет извлекать и добавлять сразу несколько элементов?

*ConcurrentStack*

6. Какой статус будет у потока при корректном завершении?

*Stop*

7. Какая из ниже перечисленных потокобезопасных коллекции представляет неупорядоченную коллекцию объектов?

*ConcurrentBag*

8. Какого приоритета потока НЕ существует?

*Bigger*

9. В каком статусе находится только что созданный поток?

*Unstarted*

10. Какой из механизмов синхронизации потоков может ограничивать количество потоков, способных одновременно обращаться к ресурсу?

*Semaphore*

11*.* Какая из ниже перечисленных потокобезопасных коллекции имеет методы вставки и изменения с приставкой "Тry'?

*ConcurrentDictionary*

12*.* Какого из этих подходов нет в параллельном программировании?

*Подход основанный на параллелизме потоков*

13*.* Что такое поток?

*наименьшая единица обработки*

14*.* Какая из этих характеристик не относится к параллельному программированию?

*уменьшение времени разработки программы*

15*.* Какая из ниже перечисленных потокобезопасных коллекции позволяет извлекать и добавлять сразу несколько элементов?

*ConcurrentStack*

Тест 6

1. Что такое async?

*Модификатор метода, указывающий, что данный метод может выполнятся асинхронно*

2. Каким способом модно запустить Task синхронно?

*Вызвать у объекта Task метод RunSynchronously*

3. Какой подход предпочтителен, если требуется получить результат из объекта Task?

*У объекта Task вызвать метод GetAwaiter и у него вызвать метод GetResult*

4. К какому типу данных относится ValueTask?

*Значимый, так как это структура*

5. В каких случаях не следует применять асинхронный подход?

*Для решения задач, связанных с вводом данных от пользователя*

6. Какой из этих вариантов не запускает автоматически задачу на выполнение при создании объекта класса Task?

*При создании через конструктор класса Task*

7. В каком случае будет вызвано исключение типа AggregateException, если в выполняемой операции возникло исключение?

*Если использовался метод Wait или свойство Result*

8. Что такое асинхронное программирование?

*Механизм выполнения ряда методов в неблокирующем режиме*

9. Какой возвращаемый тип данных менее предпочтителен для асинхронных методов?

*void*

10. В каком случае предпочтительно использовать ValueTask вместо Task?

*Если ожидается, что операция будет выполняться синхронно*

11. Что такое Task?

*Класс, предоставляющий отдельную операцию, которая выполняется асинхронно*

12. Какой из этих методов/свойств не будет блокировать выполнение основного потока , пока не завершится фоновый?

*ContinueWith*

13. Что не может возвращать элемент ValueTask?

*void*

14. Что такое конечный автомат?

*Это математическая абстракция, модель, которая может находиться только в одном из конечного числа состояний в каждый конкретный момент времени*

15. Что такое асинхронное программирование?

*Механизм выполнения ряда методов в неблокирующем режиме*

16. Что такое await?

*Унарный оператор, указывающий, что во время работы необходимо дождаться окончания выполнения операции*

17. В каком случае будет вызвано исключение типа AggregateException, если в выполняемой операции возникло исключение?

*Если использовался метод Wait или свойство Result*

Тест 7

1. Что понимается под глобализацией приложения?

*проектирование и программирование приложения, не зависящего от языка и региональных параметров и поддерживающего локализованные пользовательские интерфейсы и региональные данные для всех пользователей*

2. Как следует выполнять сравнение строк?

*Посимвольно*

3. Что такое UTC?

*Стандарт, по которому общество регулирует часы и время*

4. В каких случая НЕ требуется ручная обработка (свой код) данных?

*При работе с валютами*

5. Зачем нужен "менеджер ресурсов"?

*Для получения значений ресурсов под нужную локаль в коде приложения*

6. Что из ниже перечисленного требует локализации?

*Всё вышеперечисленное*

7. Как следует обрабатывать строки при сортировке или поиске подстроки?

*Целиком*

8. Как правильно выполнять арифметические операции с датами?

*Привести к формату UTC, выполнить операцию, привести из формата UTC в региональный*

9. Какие проблемы можно обнаружить при тестировании приложения под локализации?

*Ошибки при сохранении числовых или временных (даты и время) значений в БД*

10. Какой этап НЕ входит в процесс локализации приложения?

*Анализ глобализации*

11. Какие проблемы могут быть при полном выводе даты в региональном формате?

*Название недели и месяца могут выводить неправильно*

12. Что значит “нейтральный” ресурс

*Ресурсы, которые используются, в случае отсутствия ресурсов под нужную локаль*

13. В каком формате рекомендуется хранить строки?

*Юникод*

14. Какая проверка не входит в анализ локализуемости?

*Выполнена ли локализация приложения?*

15. Что такое “ресурс”?

*Любые неисполняемые данные, которые логически развертываются вместе с приложением*

Тест 8

1. Какой из описанных способов получения объекта типа Type является не верным?

*Через оператор GetType*

2. Как получить перечень всех методов класса через Type?

*вызывать у объекта Type метод GetMethods и получить массив объектов типа MethodInfo*

3. Какой из перечисленных этапов лишний при создании объекта от универсального (generic) класса?

*Через метод CreateInstance получить новый экземпляр для универсального типа*

4. Как извлечь значение из закрытого поля класса?

*Через объект Type от класса получить нужное поле и вызвать у него метод GetValue, передав в него объект от класса*

5. Как не получится получить информацию об атрибутах класса?

*Через метод GetCustomAttributes класса Activator*

*6. Что такое рефлексия?*

*процесс выявления типов во время выполнения приложения*

*7. Что такое BindingFlags?*

*Флаги, управляющие привязкой и способом, используемым рефлексией при поиске членов и типов*

*8. Как не получится вызать на выполнение метод класса*

*Через метод Invoke класса Activator*

*9. Какой класс является базовым классом описания элементов исследуемого типа?*

*MemberInfo*

*10. Что из этого обязательно при создании класса-атрибута?*

*Унаследовать новый класс от класса System.Attribute*

*11. Какой из перечисленных классов не является основным классов рефлексии?*

*Class*

*12. Как называется класс, через методы которого создаются объекты?*

*Activator*

*13. Как настроить класс атриубта так, чтобы он применялся только к свойствам класса?*

*Прописать у класса-атрибута атрибут AttributeUsage с указанием в свойстве ValidOn значение AttributeTargets.Property*

*14. Что нужно, чтобы занести в свойство класса значение?*

*Получить через объект Type нужное свойство, вызвть у него метод SetValue, передав туда объект класса и значение*