Вариант 2

Теоретические вопросы

1. Чем отличается модель быстрой разработки приложений от инкрементной модели?
2. Укажите сходства и различия спиральной модели и классического жизненного цикла
3. Чем отличается компонентно-ориентированная модель от спиральной модели и классического жизненного цикла?
4. Что такое конструктивная модель стоимости? Для чего она применяется?

Тесты

**5. При использовании какого метода тестирования реализация системы недоступна тестировщикам:**

а) при использовании метода белого ящика

б) при использовании любого метода тестирования

в) при использовании метода черного ящика

г) при использовании метода серого ящика

**6. Модель жизненного цикла программного обеспечения**

а) определение определенных действий, которые сопровождают изменения состояний объектов;

б) типичная схема последовательности работ на этапах разработки программного продукта;

в) отражение динамики изменений состояния каждого класса объектов;

г) моделирование процессов разработки ПО

**7. Какие возвраты невозможны при разработке по водопадной модели:**

[а) возврат от кодированию к тестированию

б) возврат от тестирования к анализу

в) возврат от тестирования к кодированию

г) возможны все возвраты

 8. **Укажите порядок приведенных ниже этапов жизненного цикла ПО**

a.             тестирование

b.            кодирование

c.             сбор требований

d.            анализ требований

e.             проектирование

f.              сопровождение

9. **Какая пара групп  требований к ПП является наиболее противоречивой**

a.             Функциональность и юзабилити

b.            Сопровождаемость и мобильность

c.             Надежность и эффективность

d.            Сопровождаемость и надежность

10. **Что несправедливо относительно инкрементной модели**

a.             основная идея – идея прототипирования

b.            разработка прототипа (инкремент) в инкрементной модели происходит в рамках этапов каскадной модели

c.             первый прототип реализует базовые требования к ПО

d.            требования заказчика уточняются при разработке каждого прототипа

e.             план следующего инкремента предусматривает модификацию предыдущего прототипа, обеспечивающую дополнительные характеристики и функциональность ПО

11. **Экспертиза программного обеспечения проводится в рамках процесса**:

1. Сопровождения;
2. Тестирования;
3. Верификации;
4. Проектирования;
5. Валидации

**12. Количественные показатели надежности включают**

1. среднее время восстановления после отказа

2. интенсивность отказа

3. коэффициент готовности

4. количество ошибок в программе в единицу времени

5. все вышеперечисленное

**13. Последовательность работ по инкрементной модели разработки программного обеспечения**:

1. Анализ, тестирование, кодирование, проектирование;
2. Анализ, кодирование, проектирование, тестирование;
3. Анализ, проектирование, тестирование, кодирование
4. Анализ, проектирование, кодирование, моделирование;
5. Анализ, проектирование, кодирование, тестирование;

**14. Модели надежности – это:**

1. модели, предназначенные для оценки надежности программ;

2. математические модели, построенные для оценки зависимости надежности программного средства от заранее известных или оцененных в ходе тестирования параметров.

3. физическая модель, определяющая степень надежности программного средства.

**15. Модель Липова относится к ……… моделям надежности программных** средств

1. аналитическим, динамическим, непрерывным;

2. статическим, по области ошибок;

3. аналитическим, динамическим, дискретным;

4. динамическим, по области ошибок.

**16. Модель Нельсона относится к … моделям надежности программных средств**

1. аналитическим, динамическим, непрерывным;

2. статическим, по области ошибок;

3. аналитическим, динамическим, дискретным;

4. динамическим, по области ошибок.

**17. Техническая экспертиза ПО это:**

1. систематический анализ артефактов проекта квалифицированными специалистами для оценки их внутренней согласованности, точности, полноты, соответствия стандартам и принятым в организации процессам, а также соответствия друг другу и общим задачам проекта.
2. метод экспертизы, в рамках которого один из членов команды проверки представляет ее участникам последовательно все характеристики проверяемого артефакта, а они анализируют его, задавая вопросы, внося замечания, отмечая возможные ошибки, нарушения стандартов и другие дефекты
3. последовательное изучение характеристик артефакта, обычно следующее некоторому плану, с целью обнаружения в нем ошибок и дефектов.
4. анализ артефактов и процессов жизненного цикла, выполняемый людьми, не входящими в команду проекта, для оценки соответствия этих артефактов и процессов задачам проекта, заключенному контракту, общим стандартам, друг другу и пр.

**18. На каком уровне требований осуществляется формализация требований**

1. уровень пользователя;
2. Верхний уровень;
3. Функциональный уровень;
4. Абстрактный уровень;
5. Системный уровень

**19. Какие из требований к ПО не используются в ГОСТ 34.602-89**

1. требования к системе в целом;
2. требования к функциям (задачам), выполняемым системой;
3. требования к видам обеспечения;
4. требования к надежности системы;
5. требования к срокам выполнения проекта
6. **Провести расчет размерно-ориентированных метрик для следующих проектов**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Проект | Затраты, чел.-мес | Стоимость, тыс. $ | KLOC, тыс. LOC | Прогр. док- ты, страниц | Ошибки | Люди |
| Bbbb1 | 36 | 239 | 11,4 | 465 | 34 | 4 |