|  |  |
| --- | --- |
| Анализируемость является составляющей характеристики | Удобство сопровождения |
| Аналитическое моделирование надежности включает | 4 шага |
| Важности дефекта | Блокирующий  Критический  Значительный  Незначительный  Тривиальный |
| Версии комплекса программ можно разделить на | Эталонные и пользовательские |
| Весь интервал времени от момента зарождения идеи о том, чтобы создать или приобрести программную систему для решения определенных задач, до момента полного прекращения использования последней ее версии | Жизненного цикла |
| Градации важности дефекта | Блокирующая ошибка – ошибка, приводящая приложение в нерабочее состояние, в результате которого дальнейшая работы с тестируемой системой или ее ключевыми функциями становится невозможна. Решение проблемы необходимо для дальнейшего функционирования системы  Критическая ошибка – неправильно работающая ключевая бизнес логика, дыра в системе безопасности, проблема, пришедшая к временному падению сервера или приводящая в нерабочее состояние некоторую часть системы, без возможности решения проблемы, используя другие входные точки. Решение проблемы необходимо ля дальнейшей работы с ключевыми функциями тестируемой системой.  Значительная ошибка – часть основной бизнес логики работает некорректно. Ошибка некритична или есть возможность для работы с тестируемой функцией, используя другие входные точки.  Незначительная ошибка – ошибка, не нарушающая бизнес логику тестируемой части приложения, очевидная проблема пользовательского интерфейса.  Тривиальная ошибка – ошибка, не касающаяся бизнес логики приложения, плохо воспроизводимая проблема, малозаметная по средствам пользовательского интерфейса, проблема сторонних библиотек или сервисов, проблема, не оказывающая никакого влияния на общее качество продукта. |
| Градация приоритета дефекта | Высокий – ошибка должна быть исправлена как можно быстрее, т.к. ее наличие является критической для проекта.  Средний – ошибка должна быть исправлена, ее наличие не является критичной, но требует обязательного решения.  Низкий - ошибка должна быть исправлена, ее наличие не является критичной, и не требует срочного решения. |
| Динамический вычислительный процесс обработки данных является | Объектом уязвимости, влияющим на надежность ПС |
| Для оперативной защиты при отказовых ситуациях наибольшее внимание уделяется | Временной избыточности |
| Для реализации фильтров, обеспечивающих повышение надежности функционирования программ и защиту вычислительного процесса программно-алгоритмическими методами, используется | Три вида избыточности |
| Дублирование накопленных исходных и промежуточных данных, обрабатываемых комплексом программ, образует | Информационную избыточность |
| Если разработчик теста имеет доступ к исходному коду программ и может писать код, который связан с библиотеками тестируемого ПО, то такое тестирование называется | Тестирование «белого ящика» |
| Если тестироващик имеет доступ к ПО только через те же интерфейсы, что и заказчик или пользователь, то такое тестирование называется | Тестированием «черного ящика» |
| Информация, выдаваемая потребителям | Объектом уязвимости, влияющим на надежность ПС |
| Искажения в каналах телекоммуникации информации, передаваемой потребителям являются | Внешними дестабилизирующими факторами |
| Искажения в каналах телекоммуникации информации, передаваемой потребителям являются | Внешним дестабилизирующими факторами |
| Использование некоторой части производительности ЭВМ для контроля исполнения программ и восстановления вычислительного процесса образует | Временную избыточность |
| К обработке Сбоев аппаратуры ОТНОСИТСЯ | ВСЕ КРОМЕ ИЗОЛЯЦИИ ОШИБОК |
| К Основным нефункциональным требованиям, существенным для большинства ПС и выражающих ограничения, актуальные для многих проблемных областей относятся | Все варианты ответов |
| Контроль и обеспечение достоверности наиболее важных решений по управлению и обработке информации образует | Программную избыточность |
| Критерии выделения машрутов в программе могут быть следующими | ВСЕ ВЕРНЫЕ |
| Критерии начала тестирования (все, кроме выдержки) | Готовность тестовой платформы  Наличие всей необходимой документации  Законченность разработки требуемого функционала |
| Критерии окончания тестирования (две выдержки) | Выдержан определенный период без изменения исходного кода приложения  Выдержан определенный период без открытия новых багов. |
| Майерс Г. В книге Искусство тестирование программ | Пошаговое тестирование  Монолитное тестирование |
| Методами сбора требований являются | Все варианты ответа |
| Методы изоляции ошибок относятся к | Обеспечение устойчивости к ошибкам |
| Методы используемые при модульном тестировании | По форме представления модуля  По компонентам программы  По степени автоматизации  Статические или динамические методы |
| Методы отступления или сокращенного обслуживания относятся к | Обеспечению устойчивости к ошибкам |
| Методы, позволяющие свести сложность программы к минимуму относятся к | Предупреждению ошибок |
| Модель переходных вероятностей | Основа на марковском процессе |
| Модель Шика-Волвертона относится к | Динамическим моделям дискретного времени |
| Модульное тестирование это - | ТОЛЬКО ВНУТРЕННИЕ РАБОЧИЕ ЧАСТИ ПРОГРАММЫ |
| Модульное тестирование, как правило осуществляется | Программистом |
| Можно выделить следующие недостатки монолитного тестирования | Все кроме ТРЕБУЕТСЯ МНОГО ВРЕМЕННЫХ ЗАТРАТ НА НАПИСАНИЕ ДРАЙВЕРОВ |
| Набор действий, направленных на решение одной задачи или группы тесно связанных задач в рамках разработки и сопровождения ПО | Понятие видов деятельности в жизненном цикле ПО |
| Насколько выбранные формы организации, планы и методы выполнения работ соответствуют задачам, решаемым в рамках проекта, и ограничениями по сроками бюджету проверяют | При верификации организационных документов |
| Неправильно работающая ключевая бизнес логика, дыра в системе безопасности, проблема, пришедшая к временному падению сервера или приводящая в нерабочее состояние некоторую часть системы, без возможности решения проблемы, используя другие входные точки. | Критическая ошибка |
| Объектный код программы является | Объектом уязвимости, влияющим на надежность ПС |
| Осуществимость требований с помощью технологий, взятых на вооружение в проекте и в рамках выделенных на проект ресурсов проверяется | При верификации описания требований |
| Ответьте нефункциональные требования к системе | Несанкционированный доступ к системе |
| Отклонение программы от функционирования или невозможность программы выполнять функции, определенные требованиями и ограничениями, рассматриваемыми как событие, способствующее переходу программы в неработоспособное состояние называются | Отказами |
| Отметьте методы сбора требований | Наблюдение за работой действующей системы с целью отделения ее проблемных свойств от тех, которые обусловлены структурою кадров |
| Оценка надежности больших многомодульных программ, основанная на использовании имитационной модели, программа которой «засоряется» ошибками, выполняется по | Модели Миллса |
| Ошибка не касающаяся бизнес логики приложения, плохо воспроизводимая проблема малозаметная по средствам пользовательского интерефейса | Тривиальная |
| Ошибка не нарушающая бизнес логику очевидная проблема пользовательского интерфейса | Незначительная |
| Ошибка, приводящая приложение в нерабочее состояние в результате которого дальнейшая работа с тестируемой системой или ее ключевыми функциями становится невозможна решение проблемы необходимо для дальнейшего функционированиия | Блокирующая |
| Ошибки программирования являются | Внутренним источником угроз надежности |
| Ошибки разработчика на любом из этапов разработки ПО, которые могут содержаться в исходных или проектных спецификациях, текстах кодов программ, эксплуатационной документации и т.п., являются | Дефектами. |
| Поведение каждого из видов динамического тестирования состоит из следующих этапов | Собственно тестирование  Обработка результатов тестирования  Планирование тестирования |
| Повышения интенсивности искажений исходных данных, а также увеличение загрузки комплекса программ выше нормальной образуют группу | Форсированных методов определения показателей надежности |
| Покрытие операторов | 205 |
| Покрытие операторов | Нет правильного ответа |
| Покрытие решений | 303 211  5010 311 |
| Покрытие условий | 204 111 |
| Понятие артефакта | Различные информационные сущности, документы и модели, создаваемые или используемые в ходе разработки и сопровождения ПО |
| Понятие валидации | Соответствие любых создаваемых или используемых в ходе разработки и сопровождения ПО артефактов нуждам и потребностям пользователей и заказчиков этого ПО, с учетом законов предметной области и ограничений контекста использования ПО |
| Понятие верификации | Соответствие одних создаваемых в ходе разработки и сопровождения ПО артефактов другим, ранее созданным или используемым в качестве исходных данных, а также соответствие этих артефактов и процессов их разработки правилам и стандартам |
| Понятие видов деятельности в жизненном цикле ПО | Набор действий, направленных на решение одной задачи или группы тесно связанных задач в рамках разработки и сопровождения ПО |
| Понятие жизненного цикла | Весь интервал времени от момента зарождения идеи о том, чтобы создать или приобрести программную систему для решения определенных задач, до момента полного прекращения использования последней ее версии |
| При введении избыточности, сокращение ресурсов ЭВМ, доступных для выполнения основных функций системы, характеризуется | Коэффициентом простоя |
| При планировании тестирования структуры программных модулей решаются 2 задачи | Формирование критериев выделения маршрутов в программе  Выбор стратегий упорядочения выделенных маршрутов |
| При тестировании классов можно выделить 5 оцениваемых факторов | В каких объемах тестировать  Что тестировать  Каким образом тестировать  Когда тестировать  Кто выполняет тестирование |
| Привлекательность является составляющей характеристики | Удобство использования |
| Примеры видов деятельности | Анализ предметной области  Выделение и описание требований  Проектирование  Разработка кода  Тестирование  Управление конфигурациями  Развертывание. |
| Приоритеты дефекта | Высокий  Средний  Низкий |
| Проведение тестирования двумя группами программистов предусмотрено в | Простой интуитивной модели |
| Провести классификацию | В разделе «Что надо тестировать?» приводятся Описания объекта тестирования  В разделе «Что будет тестироваться?» приводятся Описание тестируемой системы Описание компонент тестируемой системы в отдельности Список функций тестируемой системы  В разделе «Как будет тестироваться?» приводятся Описания видов тестирования и их применения по отношению к тестируемому объекту  В разделе «Когда будет тестироваться?» приводятся Описание последовательности проведения и перечень работ по тестированию |
| Провести классификацию | К стратегиям «белого ящика» относятся  Покрытие операторов  Покрытие решений  Покрытие условий  Покрытие решений/условий  Комбинаторное покрытие условий  К стратегиям «черного ящика» относятся  Эквивалентное разбиение  Анализ граничных значений  Анализ причин и следствий  Предугадывание ошибки  Исчерпывающее тестирование |
| Процесс проверки правильности спецификаций требований на их соответствие, непротиворечивость, полноту и выполнимость, а также на соответствие стандартам называется | Среди предложенных вариантов нет верного |
| Процесс формализованного описания функциональных и нефункциональных требований называется | Спецификацией требований |
| Процесс формализованного описания функциональных и нефункциональный требований называется | Спецификацией требований(Но в тестах ничего не выбрано) |
| Различные информационные сущности, документы и модели, создаваемые или используемые в ходе разработки и сопровождения ПО | Артефакт |
| Расположите стратегии тестирования в порядке повышения качества тестирования | Покрытие операторов  Покрытие решений  Покрытие условий  Покрытие решений/условий |
| Сбои и отказы в аппаратуре являются | Внешним источником угроз надежности |
| Содержит ли документация полное, точное и непротиворечивое описание поведения системы проверяют | При верификации пользовательской документации |
| Соедините | А = 2, В = 0, Х = 4 - 1, 5  А = 2, В = 1, Х = 1 - 2, 6  А = 1, В = 0, Х = 2 - 3, 7  А = 1, В = 1, Х = 1 - 4, 8 |
| Соответствие любых создаваемых или используемых в ходе разработки и спопровождения по артефактов нуждам и потребностям | Валидации |
| Соответствие одних создаваемых в ходе разработки и сопровождения ПО артефактов другим, ранее созданным или используемым в качестве исходных данных, а также соответствие этих артефактов и процессов их разработки правилам и стандартам | Верификации |
| Соответствие стандартам является составляющей характеристики | ВСЕ ВАРИАНТЫ ОТВЕТОВ |
| Соответствует ли описанное в документации поведение реальному поведению системы проверяют | При верификации пользовательской документации |
| Состояние программы, при котором выдается неправильные результаты, причиной которых являются изъяны в операторах программы или в технологическом процессе ее разработки, что приводит к неправильной интерпретации исходной информации, а, следовательно, и к неверному решению, называется | Ошибкой |
| Способна ли система работать в том окружении, в котором она нужна пользователям, проверяют | При верификации самой работающей системы |
| Существует | 6 видов отказовых ситуаций |
| Существует следующее количество состояний, характеризующих процесс функционирования комплекса программ на однопроцессорной ЭВМ | 5 |
| Тестирование интегрированной системы на ее соответствие исходным требованиям называется | Системным тестированием |
| Тестирование интерфейсов и взаимодействий между компонентами, называется | Интеграционным тестированием |
| Тестирование минимально возможного для тестирования компонента, например, отдельного класса или функции называется | Модульным тестированием |
| Тестирование ПО в целях проверки способности ПО в определенных условиях решать задачи, нужные пользователям называется | Функциональным тестированием |
| Тестирование ПО с целью оценки удобство использования продукта называется | Среди предложенных вариантов нет верного |
| Тестирование программного обеспечений, выполняемое на полной, интегрированной системе, с целью проверки соответствия системы исходным требованиям называется | Системным тестированием |
| Тестирование, выполняемое после внесения изменений в очередную версию программы, называется | Регрессионным тестированием |
| Точность является составляющей характеристики | Функциональность |
| Укажите составляющие характеристик качества ПО | Функциональность (Способность к взаимодействию, Защищенность, Точность, Функциональная пригодность)  Надежность (Зрелость, Устойчивость к отказам, Способность к восстановлению)  Производительность (Временная эффективность, Эффективность использования ресурсов)  Удобство сопровождения (Анализируемость, Удобство внесения изменений, Стабильность, Удобство проверки)  Удобство использования (Удобство работы, Понятность, Удобство обучения, Привлекательность)  Переносимость(Адаптируемость, Удобство установки, Способность к сосуществованию, Удобство замены) |
| Упорядочить важности дефекта | Блокирующий  Критический  Значительный  Незначительный  Тривиальный |
| Установите верную последовательность. Перечислите в правильной последовательности методику использования тестирования модулей | Ручное тестирование  Символическое тестирование  Тестирование структуры  Тестирование обработки данных  Функциональное тестирование |
| Установите соответствие между характеристиками качества ПО и их составляющими | Функциональность – Защищенность  Надежность – Зрелость  Удобство использования – Понятность Производительность – Эффективность использования ресурсов  Удобство сопровождения – Стабильность  Переносимость – Удобство установки |
| Установите соответствия | Способность решать задачи,  способность предоставлять определенные резульаты,  способность обеспечивать необходимо низкий уровень риска,  способность приносить удовлетворение |
| Установите соответствия | Бизнес требования – определяют назначение ПО  Бизнес правила – определяют ограничения  Пользовательские требования - определяют набор пользовательских  Системные требования и ограничения – определение набора элементареных |
| Установить соответствие между характеристиками качества ПО и их составляющими | Защищенность,  Зрелость,  Понятность,  Эффективность Использования ресурсов,  Стабильность,  Удобство Установки |
| Формулирует ли проектные документы все проектные решения точно и полно проверяют | При верификации проектных решений |
| Часть основной бизнес логики работает некорректною Ошибка некритична или есть возможность для работы с тестируемой функцией | Значительная |
| Шкала Позволяющая ранжировать некоторые характеристики программного продукта путем сравнения с опорными значениями называется | порядкойвой |
| Шкала, используемая для измерения реальных физических величин, называется | Интервальной |
| Шкала, которая характеризует только наличие рассматирваемомго свойства или признака у программы без учета градации по численным значениям называется | категорийной |