

Тестовые задачи

Общие вопросы:

1. По вашему - кто такой QA инженер?
2. Что такое тестирование?
3. Зачем вообще проводить проверку ПО?

Ответы:

QA инженер - это специалист по функциональному тестированию программного обеспечения (ПО) на этапе разработки.

Тестирование - это проверка программы на то, что она работает так, как планировалось.

Проводить проверку ПО необходимо, чтобы избежать рисков и поломок на этапе разработки и релиза. Это позволяет выявить ошибки на ранних этапах и подтвердить соответствие требованиям.

Логическая задача:

5 пиратов собираются разделить 100 золотых монет. Порядок пиратов определен – известно кто 1й 2й и так далее. По традиции пираты делят деньги следующим образом. 1й пират предлагает вариант деления денег. Все голосуют за или против этого варианта. Если голосов против более половины пирата убивают и следующий делит по тем же правилам. Когда дело касается денег каждый пират отлично умеют считать и поддержит предложенный вариант если он для него выгоднее чем альтернатива. Как 1му пирату получить максимально возможное количество денег и остаться в живых.

Ответ:

Чтобы первый пират получил как можно больше монет и остался в живых, ему нужно предложить план, который будет выгоден большинству пиратов.

1. Если остался только один пират, он забирает все монеты.
2. Если остались два пирата, второй заберет все монеты себе.
3. Если остались три пирата, третий предложит пирату 5 одну монету, чтобы получить его голос, и заберет остальное себе.
4. Если остались четыре пирата, второй предложит пиратам 4 и 5 по одной монете, чтобы получить их голоса, и заберет остальное себе.

5. Если все пять пиратов, первый предложит пиратам 3, 4 и 5 по одной монете, чтобы получить их голоса, и заберет себе 97 монет.

Таким образом, первый пират предложит:

- Себе: 97 монет
- Пирату 2: 0 монет
- Пирату 3: 1 монету
- Пирату 4: 1 монету
- Пирату 5: 1 монету

Пираты 3, 4 и 5 согласятся с этим, потому что это лучшее предложение для них. В итоге первый пират останется в живых и получит 97 монет.

Порядок распределения монет в данной задаче определяется стратегией, которая учитывает желание каждого пирата выжить и получить как можно больше монет.

Первый пират предлагает себе 97 монет, что является максимально возможным количеством монет, которые он может получить, сохраняя при этом поддержку остальных пиратов.

Пирату 2 не предлагается ни одной монеты, потому что если он не согласится с этим предложением, то его монеты пойдут другим пиратам в последующих ситуациях.

Пираты 3, 4 и 5 получают по одной монете каждый. Это делается для того, чтобы они поддержали предложение первого пирата. Если бы первый пират предложил им меньше монет, они бы отказались от его предложения.

Задачи на тестирование:

1. Как протестировать сломанный тостер?

Ответ :

Для данного тестирования я бы провела smoke тестирование, позитивное тестирование, негативное тестирование и юзабилити тестирование.

Исходные данные: Имеется сломанный тостер. Описанных требований нет.

Smoke тестирование:

1. Подключить тостер к электрической розетке.
2. Включить в сеть.
3. Нажать на кнопку включения.
4. Дождаться включения тостера, когда загорится индикатор.
5. Выключить тостер.

Ожидаемый результат: Тостер должен включиться, индикатор питания должен загореться.

Позитивное тестирование

Проверка нагрева:

1. Включить в сеть.
2. Нажать кнопку включения или опустить рычаг на тостере.

Ожидаемый результат: Тостер начнет нагреваться, внутренние нагревательные элементы должны засветиться.

Проверка таймера:

1. Положить ломтик хлеба в тостер.
2. Включить в сеть.
3. Нажать кнопку включения или опустить рычаг на тостере.
4. Установить таймер на среднее значение и начать процесс тостинга.

Ожидаемый результат: Тостер должен работать в течение установленного времени и автоматически отключиться, подняв хлеб.

Проверка механизма подъема:

1. Положить ломтик хлеба в тостер.
2. Включить в сеть.
3. Нажать кнопку включения или опустить рычаг на тостере.
4. Установить таймер на среднее значение и начать процесс тостинга.

5. Проверить, поднимается ли хлеб автоматически после завершения тостинга.

Ожидаемый результат: Хлеб должен подняться автоматически после завершения тостинга.

Проверка равномерности поджаривания:

1. Положить ломтик хлеба в тостер.
2. Включить в сеть.
3. Нажать кнопку включения или опустить рычаг на тостере.
4. Установить таймер на среднее значение и начать процесс тостинга.

Ожидаемый результат: Хлеб должен прожариться равномерно с обеих сторон.

Негативное тестирование:

Перегрузка тостера (проверка защиты от перегрузки):

1. Положить в тостер несколько ломтиков хлеба больше максимально рекомендованного количества.
2. Включить тостер в сеть.
3. Нажать кнопку включения.

Ожидаемый результат: Тостер должен либо автоматически отключиться (если предусмотрена защита от перегрузки), либо произойти перегрев.

Юзабилити тестирование:

1. Проверить, что кнопки для работы с тостером удобно расположены.
2. Проверить, что кнопки с противоположным эффектом не расположены слишком близко, например, кнопка увеличения желаемой температуры и сброс настроек.

3. Проверить, что в тостер удобно уложить ломтики хлеба и что хлеб находится внутри тостера.
4. Проверить, что интерфейс тостера интуитивно понятен.

В приложении к тестовому заданию вы найдете 2 скриншота - реальное приложение (приложение №1) и макет (приложение №2). Найдите ошибки при реализации. Опционально - дайте рекомендации по улучшению.

Ответ:

Если макет (приложение 2) разработал веб-разработчик по документации от заказчика, то релиз должен совпадать с макетом (приложение 1).

В данной ситуации релиз (реальное приложение) не совпадает с макетом. В реализованном макете не совпадает текст заголовка, и передан другой смысл. В макете было указано, что пользователь может забрать карту в банке по указанному адресу, а в релизе указано, что карту доставят курьером.

Также не совпадает текст и смысл в информационной части сообщения. В макете должна была быть указана информация о том, где пользователь должен забрать карту и по какому адресу банка. В релизе эта информация отсутствует.

Кроме того, в реализованном макете не совпадает кнопка. Кнопка должна быть "Готово", а в релизе кнопка "Заккрыть".

Рекомендации по улучшению

Картинка должна быть говорящей. На картинке изображен фейерверк, я бы предложила изменить заголовок на "Рады сообщить, ваша карта готова".

Информационная часть сообщения. Если это доставка карты курьером, указать временной промежуток доставки. Если карта забирается в офисе, указать время работы офиса.

Указать, какая именно карта готова.

Если у пользователя вдруг не получится забрать карту при доставке, указать контакты поддержки, куда он может обратиться в случае изменения обстоятельств.

1. Вам передали на тестирование калькулятор и список проверок к нему, которые написал предыдущий QA. Требования описаны чуть ниже. Ваша задача - проверить корректность этих проверок.

Необходимо разработать калькулятор с возможностью текстового ввода. Для пользователей было бы удобно вводить «2+2» равнозначно, как «два плюс два» или «2 плюс 2» или «два + два», и в ответ получать «4 (четыре)». При этом калькулятор может выполнить только одно действие.

Список поддерживаемых операций:

операция	слово-аналог
+	плюсс
-	минус
*	умножить на
/	разделить на

Калькулятор помимо целых чисел должен уметь считать десятичные дроби с точностью до 5 знаков после запятой. Также помимо базового расчета, пользователю должна быть доступна история выполнения им операций.

Калькулятор поддерживает числа “пи” и “е”(основание натурального логарифма).

От предыдущего тестировщика Вам достался чек-лист выполнения тестов. Очевидно, что в тестируемом приложении есть ошибка (ошибки). Локализуйте ее (их).

№	Тестовые данные	Ожидаемый результат	Фактический результат
1	1+1+1	Пользователь видит сообщение с ошибкой: Введено некорректное значение, ошибка ввода. Калькулятор может обработать только одно действие за раз.	2 (два)
2	3 разделить на 0	Ошибка математической операции. Деление на ноль не определено.	Деление на ноль не определено
3	Девять целых девять десятых плюс 0,1	10 (десять). (Так как калькулятор умеет работать с десятичными дробями с точностью до 5 знаков) Ответ можно написать как 10.0 (десять целых ноль десятых)	Данная операция не найдена
4	Пятнадцать умножить на двадцать пять	375 (триста семьдесят пять)	375 (триста семьдесят пять)
5	Одна целая две десятых разделить на семь	0.17143 (ноль целых сто семьдесят одна тысячная сорок три десятитысячных)	0,17142 (ноль целых семнадцать тысяч сто сорок два стотысячных)
6	Минус один + 1	0 (ноль)	-1 (минус один)
7	Триста четырнадцать минус 14	Должно отобразиться сообщение, о том что ввод некорректный, так как пользователь ввел значение с ошибкой.	286 (двести восемьдесят шесть)
8	пи умножить на 100	314,159 (триста четырнадцать целых сто пятьдесят девять тысячных)	314,159 (триста четырнадцать целых сто пятьдесят девять тысячных)

В условиях задания указан список поддерживаемых операций.

При вводе знака "+" текстом указано как "плюсс".

В поле номер 6:

Если вводить знак "+" как "плюс", пользователь получит ошибку, так как это будет считаться некорректным вводом данных.