STAR Homework 4: Testing in SE + MBT with LTS

Exercise 1 (HOMEWORK: White-box testing) Consider the following program, that given two integers a and b returns `true` if $a, b > 1$ and a has no square divisor $\geq b$.

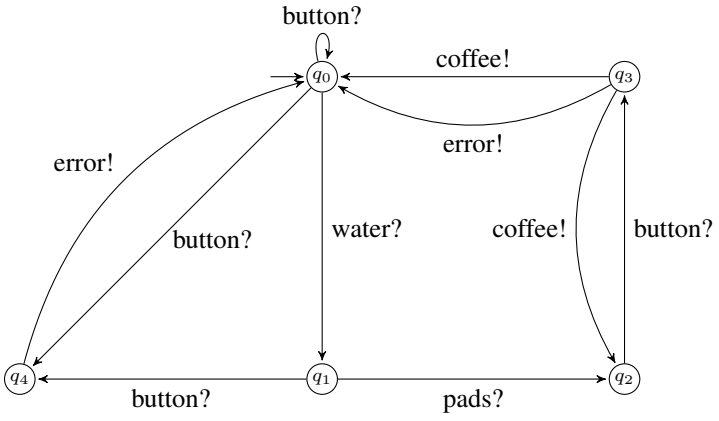
```
1  NoBigSquares(a, b :: Int) :: Bool {
2    if (a < 1 || b < 1) {
3      return false;
4    }
5    var i :: Int := ceil(sqrt(b));
6    while (i^2 =< a) {
7      if (a mod i^2 == 0) {
8        return false;
9      }
10     i := i + 1;
11   }
12   return true;
13 }
```

(a) Give a minimum-sized test suite that provides statement coverage for `NoBigSquares`, and fill in the following table:

a	b	*Output*	2	3	5	6	7	8	10	12
⋮	⋮									

This table contains one test per row. In addition to the input and expected output, each line number with a statement has a column in which 'Yes'/'No' indicates whether the statement in that line is covered by a test or not. Apart from giving the table itself, also argue why your test suite has minimum size.

Exercise 2 (HOMEWORK: after) Consider the following LTS example of a coffee machine:



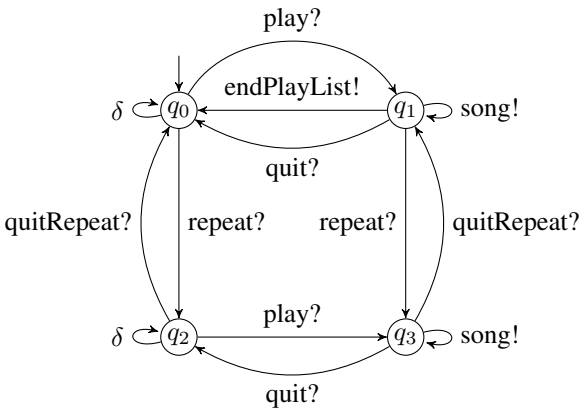
Compute the following:

(a) q_0 *after* δ button?

(b) q_3 *after* coffee! button? error!

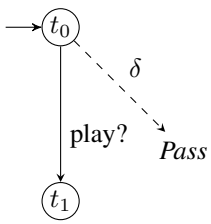
(c) q_0 *after* water? pads? coffee!

Exercise 3 (HOMEWORK: Test cases) Consider the LTS example for the Music App:

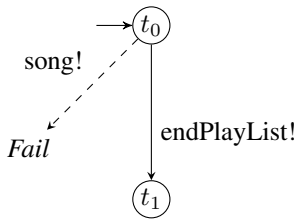


For the following test cases, indicate whether they adhere to the formal definition of a test case. If not, shortly motivate why.

(a)



(b)



Exercise 4 (White-box testing) Again, consider the following program, that given two integers a and b returns `true` if $a, b > 1$ and a has no square divisor $\geq b$.

```
1  NoBigSquares(a, b :: Int) :: Bool {
2    if (a < 1 || b < 1) {
3      return false;
4    }
5    var i :: Int := ceil(sqrt(b));
6    while (i^2 =< a) {
7      if (a mod i^2 == 0) {
8        return false;
9      }
10     i := i + 1;
11   }
12   return true;
13 }
```

(a) For this program, is it possible to have statement coverage without decision coverage?

(b) For this program, is it possible to have decision coverage without statement coverage?

(c) Is path coverage feasible for this program?

Exercise 5 (Black-box testing, equivalence partitioning) Give equivalence classes that are appropriate for the parameters of the functions below. Indicate whether the classes are valid or invalid equivalence classes given that there are two possible output errors: division by zero and taking the square root from a negative number.

(a) $\text{sign}(x) = \begin{cases} -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \\ +1 & \text{if } x > 0 \end{cases}$ where $x, \text{sign}(x) \in \mathbb{Z}$

(b) $\text{unit}(x) = \sqrt{1 - x^2}$ where $x, \text{unit}(x) \in \mathbb{R}$

(c) $\text{hyp}(x, y) = \frac{1}{x} + \frac{1}{y}$ where $x, y, \text{hyp}(x, y) \in \mathbb{R}$

(d) $\max(a, b) = \begin{cases} a & \text{if } a \geq b \\ b & \text{if } a < b \end{cases}$ where $a, b \in \mathbb{Z}$