Test Report Assignment 7

We wanted to see if we get the transitive closure from the following function:

```
trClos' :: Ord a => Rel a -> Rel a -> Rel a -> Rel a
trClos' r nxt tot = if comp `allIn` tot
                    then tot
                    else trClos' r comp (relUnion comp tot)
  where comp = r @@ nxt

trClos :: Ord a => Rel a -> Rel a
trClos r = trClos' r r r
```

We test this based on the following assumption from the workshop:

$$R \circ R \subseteq R \Leftrightarrow \text{R is transitive.}$$

From this, we designed the function testTrClos:

```
testTrClos :: [(Int, Int)] -> Bool
testTrClos rNotUnique = (r @@ r) `allIn` (trClos r)
  where r = nub rNotUnique
```

Here, the input of a random generator is converted to a unique set of elements (yet, still represented as a list), after which the stated property is tested. This function can be tested by QuickCheck using:

```
quickCheck testTrClos
```

Which returned True.