1. What property might need to hold on the result of `packetlen` for this to work? (Hint: is there a value that might be problematic for `l` to have?)

```
int divide(int numerator, int divisor);

int s=packetsum(p);
int l=packetlen(p);
int res=divide(s,l);
```

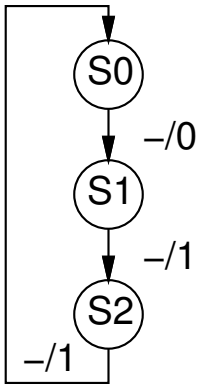
2. Assuming `findloc` is intended to return the index of `target` in the `a` array or a negative value if `target` does not exist in the first `limit` elements of `a`, what property should hold for `my_array[loc]` after the call to `findloc`?

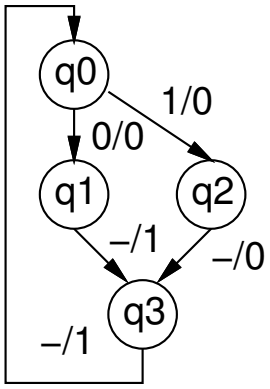
```
int findloc(int target, int *a, int limit);
.
.
.
int loc;

loc=findloc(my_target,my_array,MY_ARRAY_LEN);
// property on my_array[loc] should hold here?
```

3. Consider the following Finite State Machines:

   Start in 0 state (S0 or q0). Value before slash is input, after slash is output.
   Dash (-) means don't-care; It matches both 0 and 1.



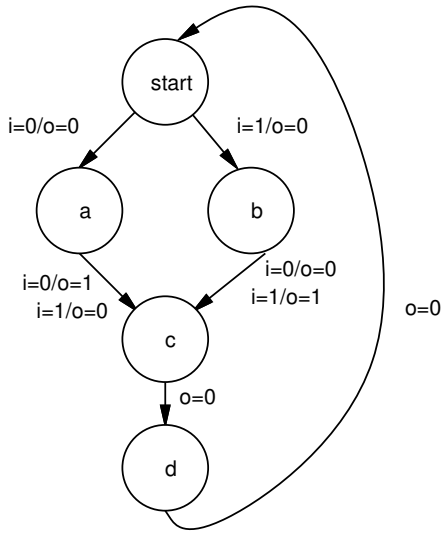


   Are the two FSMs equivalent? (*i.e.*, do they produce the same observable output sequence for any input sequence?)

   If they are not equivalent, what is an input sequence on which they behave differently?

4. Consider the following Finite State Machines (in different representations and with different number of states):

i	State	NextState	o
0	S0	S1	0
1	S0	S2	0
0	S1	S3	1
1	S1	S4	0
0	S2	S4	0
1	S2	S4	1
-	S3	S5	0
-	S4	S5	0
-	S5	S0	0



Are the two FSMs equivalent?

If they are not equivalent, what is an input sequence on which they behave differently?