

Answers examination FMSE (IA164), autumn 2008

1. see answers in separate file

2. (a) `const MONEY = 3`
`range M = 1..MONEY`

```
ENTRANCE = ( reserve -> turn_on -> get_number -> ENTRANCE
             | amount[m:M] -> pay[m] -> ENTRANCE).
```

```
CABIN = (turn_on -> (end -> amount[1] -> CABIN
                   | more -> (end -> amount[2] -> CABIN
                              | more -> end -> amount[3] -> CABIN
                              )
                   )
        ).
```

```
CUSTOMER = (reserve -> get_number -> more -> more ->
            end -> pay[m:M] -> CUSTOMER).
```

```
||SAUNA = (ENTRANCE || CABIN || CUSTOMER)\{turn_on, amount[M]}.
```

(b) `range M = 1..MONEY`
`const N_CUST = 3`
`const N_CAB = 2`
`range C = 1..N_CUST`
`range A = 1.. N_CAB`

```
ENTRANCE1 = ( c[i:C].reserve -> a[j:A].turn_on -> c[i].get_number[j] ->
ENTRANCE1).
```

```
ENTRANCE2 = ( a[j:A].amount[m:M] -> c[C].pay[j][m] -> ENTRANCE2).
```

```
CABIN = (turn_on -> (c[i:C].end -> amount[1] -> CABIN
                   | c[i:C].more -> (c[i].end -> amount[2] -> CABIN
                                       | c[i].more -> c[i].end -> amount[3] ->
CABIN
                                       )
                   )
        ).
```

```
||CABINS = a[A]:CABIN.
```

```
CUSTOMER = (reserve -> get_number[j:A] -> a[j].more ->
            a[j].end -> pay[j][m:M] -> CUSTOMER).
```

```
||CUSTOMERS = c[C]:CUSTOMER.
```

```
||SAUNA = (ENTRANCE1 || ENTRANCE2 || CABINS || CUSTOMERS)
/forall[i:C][j:A] {c[i].a[j].more/a[j].c[i].more},
forall[i:C][j:A] {c[i].a[j].end/a[j].c[i].end}}.
```

(c)

```
property REQUESTS = (c[C].reserve -> REQ[1]),
REQ[k:-3..3] = ( when(k<2) c[C].reserve -> REQ[k+1]
| c[C].pay[A][M] -> REQ[k-1]).
```

```
||SAUNA = (ENTRANCE1 || ENTRANCE2 || CABINS || CUSTOMERS || REQUESTS)
/{forall[i:C][j:A] {c[i].a[j].more/a[j].c[i].more},
 forall[i:C][j:A] {c[i].a[j].end/a[j].c[i].end}}.
```

Trace to property violation in REQUESTS:

```
c.1.reserve
a.1.turn_on
c.1.get_number.1
c.2.reserve
a.2.turn_on
c.2.get_number.2
c.3.reserve
```

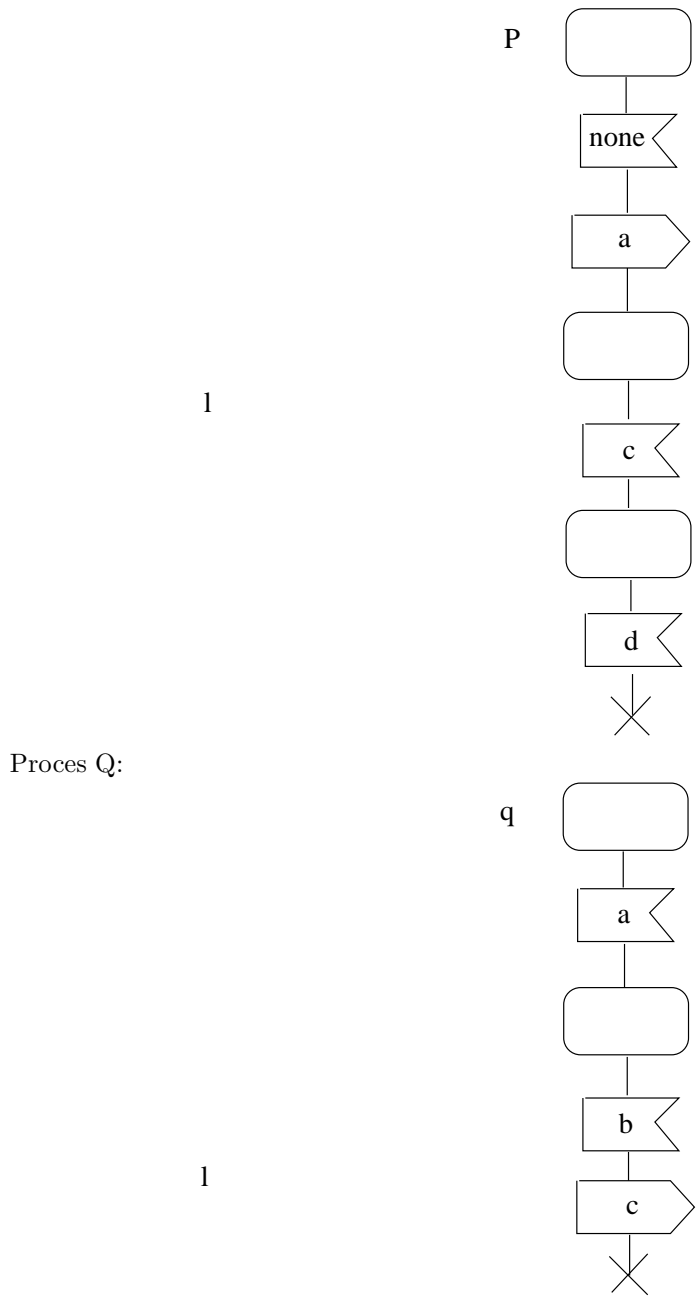
(d)

```
progress CUSTOMER3 = {c[3].a[A].end}
```

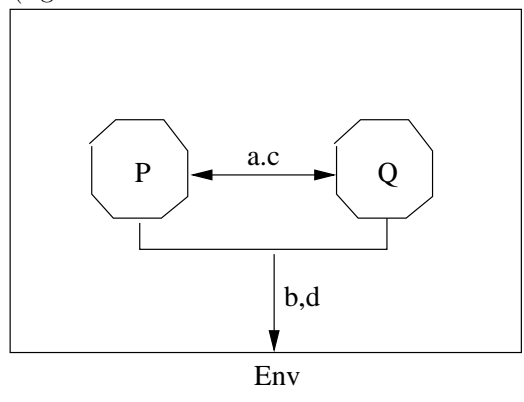
This property holds, even if we give priority to the requests of customer 1 and 2, in the following way:

```
||SAUNA = (ENTRANCE1 || ENTRANCE2 || CABINS || CUSTOMERS)
/{forall[i:C][j:A] {c[i].a[j].more/a[j].c[i].more},
 forall[i:C][j:A] {c[i].a[j].end/a[j].c[i].end}}
<<{c[1].reserve. c[2].reserve}.
```

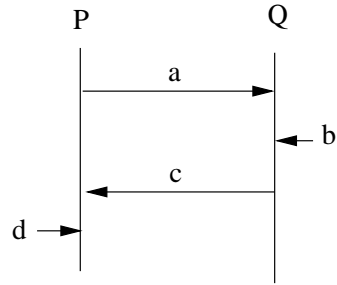
3. (a) Proces P:



(b) The SDL block (signaldefinitions could be included but I consider that not necessary):



(c) The MSC diagram:



(d) A trace of this MSC is given by

$a/@P \ /a@ \ /b@Q \ c/@Q \ /c@P \ /d@P$

with $b/@ENV$ inserted somewhere before $/b@Q$, and $d/@ENV$ inserted somewhere before $/d@P$.