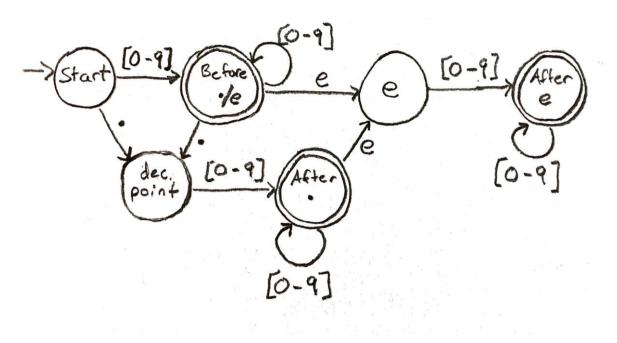
Stuart Harley

Assignment 1

FSM for a machine accepting unsigned real numbers:



Test Output:

```
n.py × 👼 turnstile.py × 🐉 fsm.py × 🐉 stepper.py × 🐉 realchecker.py
in.py × 💰 turnstile.py × 🐔 fsm.py × 🐔 stepper.py × 🐔 realchecker.py × 🐔 real.py ×
                                                                        from stepper import *
from stepper import *
                                                                        from real import *
 from real import *
                                                                        stepper = FsmlStepper(real)
 stepper = FsmlStepper(real)
 results = stepper.run(['3', '6', '.', '5', '4', 'e', '0', '9'])
                                                                         results = stepper.run(['3', '6', '.', '5', '4', 'e'])
 except FsmlIllegalEventException:
                                                                        except FsmlIllegalEventException:
    print('Illegal input')
                                                                             print('Illegal input')
 try
e realchecker
                                                                         realchecker
  C:\Users\harleys\AppData\Local\Programs\Python\Python38\python.exe
                                                                         C:\Users\harleys\AppData\Local\Programs\Python\Python38\j
  Taking step from Start on 3
                                                                         Taking step from Start on 3
  Taking step from Before ./e on 6
                                                                         Taking step from Before ./e on 6
  Taking step from Before ./e on .
                                                                         Taking step from Before ./e on .
  Taking step from Decimal point on 5
  Taking step from After . on 4\,
                                                                         Taking step from Decimal point on 5
  Taking step from After . on e
                                                                         Taking step from After . on 4
  Taking step from E on 0
                                                                         Taking step from After . on e
  Taking step from After e on 9
                                                                         Final state of machine: E
  Final state of machine: After e
                                                                         Machine REJECTED input ['3', '6', '.', '5', '4', 'e']
  Machine accepted input ['3', '6', '.', '5', '4', 'e', '0', '9']
                                                                         Machine output: []
  Machine output: []
                                                                         Process finished with exit code 0
  Process finished with exit code 0
```

```
n.py × 🐞 turnstile.py × 🐞 fsm.py × 🐞 stepper.py ×
n.py × 🐞 turnstile.py × 🐞 fsm.py × 🐞 stepper.py × 🐞 realchecker.p
 from stepper import *
                                                      from stepper import *
 from real import *
                                                      from real import *
 stepper = FsmlStepper(real)
                                                      stepper = FsmlStepper(real)
     results = stepper.run(['3', '6', '.', '5', '4'])
                                                          results = stepper.run(['3', '6', '.'])
 except FsmlIllegalEventException:
                                                      except FsmlIllegalEventException:
     print('Illegal input')
                                                          print('Illegal input')
 realchecker
  C:\Users\harleys\AppData\Local\Programs\Python\Pythor
  Taking step from Start on 3
                                                       C:\Users\harleys\AppData\Local\Programs\Pytho
  Taking step from Before ./e on 6
                                                      Taking step from Start on 3
 Taking step from Before ./e on .
                                                      Taking step from Before ./e on 6
 Taking step from Decimal point on 5
                                                       Taking step from Before ./e on .
  Taking step from After . on 4
                                                      Final state of machine: Decimal point
  Final state of machine: After .
  Machine accepted input ['3', '6', '.', '5', '4']
                                                       Machine REJECTED input ['3', '6', '.']
  Machine output: []
                                                       Machine output: []
  Process finished with exit code 0
                                                       Process finished with exit code 0
n.py × 🐞 turnstile.py × 🐞 fsm.py × 🐞 stepper.
                                                 n.py × 🐞 turnstile.py × 🐞 fsm.py × 🐞 stepper.py × 🐞 realcl
  from stepper import *
                                                  from stepper import *
                                                   from real import *
  from real import *
                                                  stepper = FsmlStepper(real)
  stepper = FsmlStepper(real)
                                                  try:
  try:
                                                   results = stepper.run(['.', '3', 'e', '8'])
  results = stepper.run(['3'])
                                                  except FsmlIllegalEventException:
  except FsmlIllegalEventException:
                                                       print('Illegal input')
      print('Illegal input')
                                                   try
                                                   realchecker X
                                                   C:\Users\harleys\AppData\Local\Programs\Python\I
 realchecker ×
                                                   Taking step from Start on .
   C:\Users\harleys\AppData\Local\Progr
                                                   Taking step from Decimal point on 3
   Taking step from Start on 3
                                                    Taking step from After . on e
   Final state of machine: Before ./e
                                                   Taking step from E on 8
   Machine accepted input ['3']
                                                   Final state of machine: After e
                                                    Machine accepted input ['.', '3', 'e', '8']
  Machine output: []
                                                   Machine output: []
   Process finished with exit code 0
```

Process finished with exit code 0

```
n.py × 🐞 turnstile.py × 🐞 fsm.py × 🐞 stepper.py
                                                     n.py 🗡 👸 turnstile.py 🗡 👸 fsm.py 🗡 👸 stepper.py 🗡 🐞 realchecker.
 from stepper import *
                                                      from stepper import *
  from real import *
                                                      from real import *
                                                      stepper = FsmlStepper(real)
 stepper = FsmlStepper(real)
 try:
                                                          results = stepper.run(['1', 'e', '8', 'e', '0'])
      results = stepper.run(['e', '8'])
                                                      except FsmlIllegalEventException:
  except FsmlIllegalEventException:
                                                          print('Illegal input')
      print('Illegal input')
                                                       realchecker ×
                                                       C:\Users\harleys\AppData\Local\Programs\Python\Pytho
  realchecker X
                                                       Taking step from Start on 1
   C:\Users\harleys\AppData\Local\Programs
                                                       Taking step from Before ./e on e
   Taking step from Start on e
                                                       Taking step from E on 8
  Illegal input
                                                       Taking step from After e on e
                                                       Illegal input
  Process finished with exit code 0
                                                       Process finished with exit code 0
n.py × 👸 turnstile.py × 👸 fsm.py × 👸 stepper.py × 👸 realchecker.
                                                       n.py × 👸 turnstile.py × 🐞 fsm.py × 🐉 stepper.py × 🐞 realchecker.py
 from stepper import *
                                                         from stepper import *
 from real import *
                                                         from real import *
 stepper = FsmlStepper(real)
                                                         stepper = FsmlStepper(real)
                                                            results = stepper.run(['1', '.', '8', 'B', '0'])
     results = stepper.run(['1', '.', '8', '.', '0'])
                                                         except FsmlIllegalEventException:
 except FsmlIllegalEventException:
                                                            print('Illegal input')
     print('Illegal input')
                                                         realchecker >
                                                          C:\Users\harleys\AppData\Local\Programs\Python\Python3
```

Taking step from Start on 1

Taking step from After . on B

Illegal input

Taking step from Before ./e on .

Taking step from Decimal point on 8

Process finished with exit code 0

C:\Users\harleys\AppData\Local\Programs\Python\Pytho

Taking step from Start on 1

Taking step from After . on .

Illegal input

Taking step from Before ./e on .

Taking step from Decimal point on 8

Process finished with exit code 0

```
.py × 🐞 turnstile.py × 🐞 fsm.py × 🐞 stepper.py × 👔
n.py × 🐞 turnstile.py × 🐞 fsm.py × 🐞 stepper.py × 🐞 realched
                                                       from stepper import *
 from stepper import *
                                                       from real import *
 from real import *
 stepper = FsmlStepper(real)
                                                       stepper = FsmlStepper(real)
 try:
     results = stepper.run(['1', 'e', '8', '0'])
                                                           results = stepper.run(['.', '5', '0'])
 except FsmlIllegalEventException:
                                                       except FsmlIllegalEventException:
     print('Illegal input')
                                                           print('Illegal input')
 realchecker ×
                                                        realchecker ×
  C:\Users\harleys\AppData\Local\Programs\Python\Py
                                                        C:\Users\harleys\AppData\Local\Programs\Py
  Taking step from Start on 1
                                                        Taking step from Start on .
  Taking step from Before ./e on e
                                                        Taking step from Decimal point on 5
  Taking step from E on 8
                                                        Taking step from After . on 0
  Taking step from After e on 0
                                                        Final state of machine: After .
  Final state of machine: After e
                                                        Machine accepted input ['.', '5', '0']
  Machine accepted input ['1', 'e', '8', '0']
                                                        Machine output: []
  Machine output: []
                                                        Process finished with exit code 0
  Process finished with exit code 0
```

```
iin.py × 🐞 turnstile.py × 🐞 fsm.py × 🐞 stepper.py
.py × 🐞 turnstile.py × 🐞 fsm.py × 🐞 stepper.py
                                              from stepper import *
from stepper import *
from real import *
                                             from real import *
stepper = FsmlStepper(real)
                                              stepper = FsmlStepper(real)
try:
                                              try:
    results = stepper.run(['.'])
                                                  results = stepper.run([])
except FsmlIllegalEventException:
                                              except FsmlIllegalEventException:
    print('Illegal input')
                                                  print('Illegal input')
 realchecker X
                                              realchecker X
 C:\Users\harleys\AppData\Local\Programs\
                                               C:\Users\harleys\AppData\Local\Progra
 Taking step from Start on .
                                               Final state of machine: Start
 Final state of machine: Decimal point
                                               Machine REJECTED input []
 Machine REJECTED input ['.']
                                               Machine output: []
 Machine output: []
 Process finished with exit code 0
                                               Process finished with exit code 0
```

```
n.py × turnstile.py × fsm.py × stepper.py ×

from stepper import *

from real import *

stepper = FsmlStepper(real)

try:

results = stepper.run(['6', 'e', '7', '.', '4'])

except FsmlIllegalEventException:
 print('Illegal input')

try

realchecker ×

C:\Users\harleys\AppData\Local\Programs\Python\Python38

Taking step from Start on 6

Taking step from Before ./e on e

Taking step from After e on .

Illegal input

Process finished with exit code 0
```

Analysis of Tests:

These inputs were accepted by the machine: 36.54e09, 36.54, 3, .3e8, 1e80, .50

These inputs were rejected by the machine because they ended in a non-final state: 36.54e, 36., . (just a decimal point), and the empty string.

These inputs threw a FsmlIllegalEventException, and so it was caught and "Illegal Input" was outputted: e8, 1e8e0, 1.8.0, 1.8B0, 6e7.4

I believe that these test cases capture every possible path through the real FSM, and each test behaves as it should. Valid inputs that ended in a final state were accepted. Valid inputs that ended in a non-final state were rejected by the machine. And non-valid inputs threw an exception which was caught and dealt with accordingly.