```
#!/usr/bin/env python
"""
Author: Michael Kepple
Date:   23 Mar 2013
Desc:   SimPy simulation of a three-teller bank
"""
from SimPy.Simulation import *
from random import expovariate, seed

# Constant values for initial simulation environment
BEGINTIME = 0
NUMTELLERS = 3
SECINSHIFT= 21600
MAXCUSTOMERS = 1000
INTERARRIVAL = 60
SERVICE = 150

class Source(Process):
    """Generates customers for simulation"""
    def generate(self,number,interarrival,tellers):
        for i in range(number):
            tempCust = Customer(name = "Customer%02d"%(i,))
            activate(tempCust, tempCust.visit(tellers))
            randomTime = expovariate(1.0 / interarrival)
            yield hold, self, randomTime

def custInTellerLine(teller):
    """Total number of customers in this teller line"""
    return (len(teller.waitQ) + len(teller.activeQ))

class Customer (Process):
    """Customer arrives, chooses shortest queue, and gets serviced"""
    def visit(self, tellers):
        arrive=now()
        QueueLength = [custInTellerLine(tellers[i]) for i in range(NUMTELLERS)]
        for i in range(NUMTELLERS):
            if QueueLength[i] == 0 or QueueLength[i] == min(QueueLength):
                choice = i
                break
        yield request,self,tellers[choice]
        queueWait = now() - arrive
        waitMon.observe(queueWait)
        waitMon.min = min(waitMon.min, queueWait)
        waitMon.max = max(waitMon.max, queueWait)
        serviceTime = expovariate(1.0 / SERVICE)
        yield hold, self, serviceTime
        yield release, self, tellers[choice]
        #serviceWait = now()-queueWait
        servMon.observe(serviceTime)
        servMon.min = min(servMon.min, serviceTime)
        servMon.max = max(servMon.max, serviceTime)

## Model/Experiment
teller = [Resource(name="Teller1"), Resource(name="Teller2"), Resource(name="Teller
3")]
servMon = Monitor()
waitMon = Monitor()
servMon.min = waitMon.min = sys.maxint
servMon.max = waitMon.max = -sys.maxint
initialize()
world = Source(name='OutsideWorld')
activate(world,world.generate(number=MAXCUSTOMERS, interarrival=INTERARRIVAL, telle
rs=teller),at=BEGINTIME)
```

```
    simulate(until=SECINSHIFT)
    print "Customers Served: %d"%servMon.count()
    print "Average Queue Time (secs): %d"%waitMon.mean()
    print "Maximum Queue Time (secs): %d"%waitMon.max
    print "Minimum Queue Time (secs): %d"%waitMon.min
    print "Average Service Time (secs): %d"%servMon.mean()
    print "Maximum Service Time (secs): %d"%servMon.max
    print "Minimum Service Time (secs): %d"%servMon.min
```