6.00 Introduction to Computer Science and Programming
Fall 2008

6.00 Handout, Lecture 19
(Not intended to make sense outside of lecture)

```
class Drunk(object):
  def __init__(self, name):
    self.name = name
  def move(self, field, cp, dist = 1):
    if field.getDrunk().name != self.name:
      raise ValueError('Drunk.move called with drunk not in field')
    for i in range(dist):
      field.move(cp, 1)

class UsualDrunk(Drunk):
  def move(self, field, dist = 1):
    cp = random.choice(CompassPt.possibles)
    Drunk.move(self, field, CompassPt(cp), dist) #Note notation of call

class ColdDrunk(Drunk):
  def move(self, field, dist = 1):
    cp = random.choice(CompassPt.possibles)
    if cp == 'S':
      Drunk.move(self, field, CompassPt(cp), 2*dist)
    else:
      Drunk.move(self, field, CompassPt(cp), dist)

class EWDrunk(Drunk):
  def move(self, field, time = 1):
    cp = random.choice(CompassPt.possibles)
    while cp != 'E' and cp != 'W':
      cp = random.choice(CompassPt.possibles)
    Drunk.move(self, field, CompassPt(cp), time)

def performSim(time, numTrials, drunkType):
  distLists = []
  for trial in range(numTrials):
    d = drunkType('Drunk' + str(trial))
  …

def ansQuest(maxTime, numTrials, drunkType, title):
  means = []
  distLists = performSim(maxTime, numTrials, drunkType)
  …

ansQuest(500, 100, UsualDrunk, 'UsualDrunk')
```

```
class oddField(Field):
  def isChute(self):
    x, y = self.loc.getCoords()
    return abs(x) - abs(y) == 0
  def move(self, cp, dist):
    Field.move(self, cp, dist)
    if self.isChute():
      self.loc = Location(0, 0)
```