analysis.py (original)

```python
# analysis.py
# ----------
# Licensing Information: Please do not distribute or publish solutions to this
# project. You are free to use and extend these projects for educational
# purposes. The Pacman AI projects were developed at UC Berkeley, primarily by
# John DeNero (denero@cs.berkeley.edu) and Dan Klein (klein@cs.berkeley.edu).
# For more info, see http://inst.eecs.berkeley.edu/~cs188/sp09/pacman.html

#######################
# ANALYSIS QUESTIONS #
#######################

# Change these default values to obtain the specified policies through
# value iteration.

def question2():
    answerDiscount = 0.9
    answerNoise = 0.2
    return answerDiscount, answerNoise

def question3a():
    answerDiscount = 0.9
    answerNoise = 0.2
    answerLivingReward = 0.0
    return answerDiscount, answerNoise, answerLivingReward
    # If not possible, return 'NOT POSSIBLE'

def question3b():
    answerDiscount =
    answerNoise = 0.
    answerLivingRewa
    return answerDiscount, answerNoise, answerLivingReward
    # If not possible, return 'NOT POSSIBLE'

def question3c():
    answerDiscount = 0.9
    answerNoise = 0.2
    answerLivingReward = 0.0
    return answerDiscount, answerNoise, answerLivingReward
    # If not possible, return 'NOT POSSIBLE'

def question3d():
    answerDiscount = 0.9
    answerNoise = 0.2
    answerLivingReward = 0.0
    return answerDiscount, answerNoise, answerLivingReward
    # If not possible, return 'NOT POSSIBLE'

def question3e():
    answerDiscount = 0.9
    answerNoise = 0.2
    answerLivingReward = 0.0
    return answerDiscount, answerNoise, answerLivingReward
    # If not possible, return 'NOT POSSIBLE'

def question6():
    answerEpsilon = None
    answerLearningRate = None
    return answerEpsilon, answerLearningRate
    # If not possible, return 'NOT POSSIBLE'

if __name__ == '__main__':
    print 'Answers to analysis questions:'
    import analysis
```

```python
for q in [q for q in dir(analysis) if q.startswith('question')]:
    response = getattr(analysis, q)()
    print '  Question %s:\t%s' % (q, str(response))
```