# Intelligent Systems (Fall 2012)

## Assignment 3: Reinforcement Learning

DUE: Nov. 30

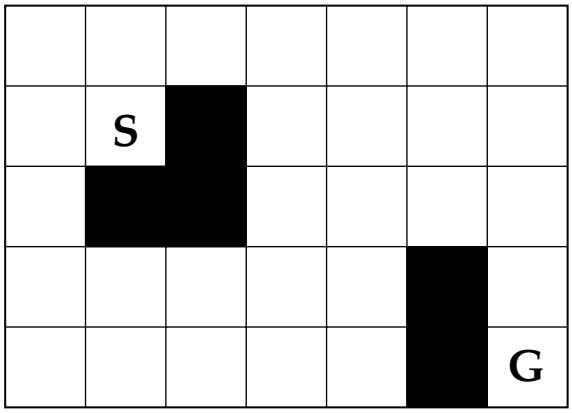


Figure 1: **30-State Maze**. For all questions refer to this figure. The agent has four possible actions: North, South, East, West. When the agent takes an action, it goes to the adjacent state in the chosen direction with probability 0.85, and in one of the other directions with probability 0.15. For example, if the agent chooses North, then there is a 85% chance that it actually goes North, a 5% chance it will go South, 5% it will go West, and 5% it will go East. If the agent goes in a direction that will take it outside the maze (e.g. going South in S), it stays in the same state. The reward $r$ is 0 for all state transitions, except that when entering the goal state **G** the reward is 10.0. The discount factor $\gamma$ is set to $0.9$. The agent cannot leave the goal state. You may number the states any way you want.

**Question 1**.

**A**. (40 points) Implement **Policy Evaluation**. Starting with $V(s) = 0, \forall s$, and assuming a random policy ($\pi(s, a) = 1/4, \forall s, a$), what are the final values, $V(s)$, after the evaluation has converged?

**B**. (10 points) How and why do the values change if the discount factor, $\gamma$ is changed to $0.7$ (again starting with $V(s) = 0, \forall s$)?

**Question 2**.

**A**. (30 points) Implement **Q-learning** with learning rate $\alpha = 0.4$. Initialize $Q(s, a) = 0, \forall s, a$. Starting each episode in state **S**, run Q-learning until it converges, using an $\epsilon$-greedy policy. Each episode ends after $100$ actions or once the goal, **G**, has been reached, whichever happens first.

**B**. (10 points) Plot the accumulated reward for the run, i.e. plot the total amount of reward received so far against the number of episodes, and show the greedy policy with respect to the value function.

**Question 3**. (BONUS 10 points) If instead of moving N,S,E,W, the agent moves like a *knight* in chess (for example, one step North, then two steps East, in one move), how would the value of the states change (run Q-learning with this new action set)?