Washington State University
School of Electrical Engineering and Computer Science
Fall 2018

CptS 440/540 Artificial Intelligence
Homework 9
Due: November 8, 2018

General Instructions: Your solution to this homework should consist of Agent.h and Agent.cc files (or a PyAgent.py file), perhaps with a readme.txt file. Put your files into one zip file, with the name <yourLastName>–<yourFirstName>.zip and submit as an attachment under Content → Homework 9 for the course CptS 440 Pullman (all sections of CptS 440 and 540 are merged under the CptS 440 Pullman section) on the Blackboard Learn system by the above deadline. Note that you may submit multiple times, but we will only grade the most recent entry submitted before the above deadline.

1. For this problem, you will add to your wumpus world agent the ability to compute the probability of a pit in each location given the evidence collected so far. The general algorithm for computing this probability is given below. The functionality of your agent will be the same as in Homework 5, except for the following changes.

    a. When the agent is faced with the choice of going forward into a location with a non-zero probability of containing a pit, the agent should do so only if the probability is less than 0.5, in which case it should visit the location with the lowest probability of containing a pit. If all unvisited locations have P(pit) > 0.5, then the agent should leave the game.
    b. Before your agent's Process method returns an action, it should output the $P(Pit_{x,y}=true)$ for all 16 locations in the Wumpus world. All test worlds will be 4x4.

    Submit your Agent.h and Agent.cc files, or PyAgent.py file, along with an optional readme.txt file containing any information you think we may need about your agent. Your agent should not require any user input. Your agent will be tested by copying only your Agent.h and Agent.cc files, or PyAgent.py file, into a fresh copy of the simulator code (that includes the new Makefile, Search.h, Search.cc and Search.py from HW5), compiling and running it on several test worlds. *The test worlds will all be 4x4, but there are no other constraints on the test worlds, e.g., there may be no way to get the gold*. Your agent will be given 10 tries on each test world. Your grade will be based on satisfying the above requirements, good programming style[1], and your agent's scores on the test worlds.

[1] Good programming style guides for C++ and Python are available on the course website.

Below is the general algorithm for computing the probability of a pit in all locations on the frontier, which consists of all unvisited locations adjacent to visited locations.

known = {information about pits in visited locations}
breeze = {information about breezes in visited locations}
frontier = {information about pits in unvisited locations adjacent to visited locations}

foreach location (x,y) in *frontier*
{
  $P(Pit_{x,y}=true) = 0.0$
  $P(Pit_{x,y}=false) = 0.0$
  frontier' = *frontier* – $\{Pit_{x,y}\}$

  foreach possible combination *C* of pit=true and pit=false in *frontier'*
  {
    P(*frontier'*) = $(0.2)^T * (0.8)^F$, where T = number of pit=true in *C*, and
      F = number of pit=false in *C*
    if *breeze* is consistent with (*C* + $Pit_{x,y}$=true)
    then $P(Pit_{x,y}=true) = P(Pit_{x,y}=true) + P(frontier')$
    if *breeze* is consistent with (*C* + $Pit_{x,y}$=false)
    then $P(Pit_{x,y}=false) = P(Pit_{x,y}=false) + P(frontier')$
  }
  $P(Pit_{x,y}=true) = P(Pit_{x,y}=true) * 0.2$
  $P(Pit_{x,y}=false) = P(Pit_{x,y}=false) * 0.8$
  $P(Pit_{x,y}=true) = P(Pit_{x,y}=true) / (P(Pit_{x,y}=true) + P(Pit_{x,y}=false))$ *// normalize*
}