Lab 0

CS 3793 – Fall 2011 assigned August 24, 2011
Tom Bylander, Instructor due midnight, September 9, 2011

In Lab 0, you will construct a program for the Pick-a-Number environment. This lab is intended for you to get accustomed to the agent software that you will be using for your lab assignments. It is worth 10 pts. compared to the 100 pts. each for the other labs.

Pick a Number

In this game, the environment picks a number between 1 to 100, and the agent guesses the number. In the regular game, the following would happen. If the guess is too low, the agent will get a “low” message. If the guess is too high, the agent will get a “high” message. If the guess is correct, the agent will get a “correct” message.

However, your lab will run with a different message from the environment. If the guess is correct, the agent will again get a “correct” message. Otherwise, the message will be 10, 20, . . . , or 100, indicating that the agent’s guess is within that distance, but not within the next lowest distance. For example, a message of 30 indicates that the guess is within 30 of the answer, but farther than 20.

Environment

The environment program for Pick-a-Number is in /home/bylander/agents/linux/bin/picknum. The source for this program and other programs are in the /home/bylander/agents/linux/src/pickanum directory. The guessnum.c program might be a useful place for you to start because it already follows the protocol. However, the binary search implemented by guessnum.c does not work for the for this case.

To run the environment for yourself, do the following:

```
setenv A /home/bylander/agents/bin or A=/home/bylander/agents/bin
$A/picknum -c
```

The -c option results in the approximate distances as described above.

The protocol for an agent program works as follows. First, the agent program will receive a message start. Then the agent sends an x message guessing that the number is x. The response to an x message is correct, 10, 20, . . . , or 100. The environment is set up to play as many times as you want, but your program should exit after the correct message.

In this lab, we want a program to read in the initial information, search for and find an optimal sequence of moves, and print out the sequence one line at a time. We can set this up using the interact program in /home/bylander/agents/linux/bin. The interact program inputs two programs on the command line, and then it reads stdout of each program and prints it to stdin of the other program. Any output to stderr will be printed to the user. There are two exceptions. If the user types in a line, this line will be printed to both programs with the prefix “user”. If the interact program reads a line that starts with “user”, then that is printed to the user and not to the programs. You do not need to understand the
interact program, but it might be an instructive exercise in learning Perl. Do not change interact. It has been carefully written so that it won’t be the cause of any deadlock.

Simple programs that might be instructive for environment/agent interaction are in /home/bylander/agents/src/picknum. Note that fflush is used after every line printed (otherwise, the programs would deadlock with unflushed buffers). In this directory, picknum is the environment, and guessnum is the agent. Running picknum -h will give you some help. Run the following to see the interaction between the two programs.

$A/interact -v $A/picknum $A/guessnum

Agent

Your task is to write a Pick-a-Number player that will do a good job at guessing the correct number for this case. From a few guesses, it should be easy to infer a small range for the correct answer. Your implementation should make no more than 10 guesses for any game.

Turning in Your Lab

Somewhere in your directory, you should create a lab0 subdirectory. This directory must have a Makefile (or must have an executable file make_lab0) that compiles and links your agent programs, which must be named lab0. Any source code that is linked to these programs must be in this directory. That is, you should be able to copy the files in your lab0 directory to another directory and be able to run the following command sequence.

setenv A /home/bylander/agents/linux/bin or A=/home/bylander/agents/linux/bin
/bin/rm lab0
source /etc/.login revert to default values for shell variables
make or ./make_lab0
$A/interact ’$A/picknum -c’ ./lab0

Zip or tar the entire directory and submit it using Blackboard. If you submit multiple times, only the last one is kept. Be sure that Blackboard has registered your submission.

In addition, provide a brief report describing the performance of your program. For example, what is the average of moves over 10 runs?