Homework assignment 2 (25 points)

Due date: Tuesday 9/29/2015

You are asked to develop a search intelligent agent which take as input two Web page

URLs and find a path of links from one to the other. In this homework, please discuss:

Start state, search space (how big), actions, and heuristics in your search, the search algorithm you will use. Please provide pseudo code to implement the search algorithm for this agent. Please work through a specific example of two specific web pages of your choices.

**Please submit your write-up on D2L HW2 dropbox by midnight of the due date.**

The start state in the search agent that finds links from one path to another would be the website that the first link is, and the collection of links on that page. The search space is going to be any link that is on the current page, and following that link and so on, so a huge amount of websites, if not the entire internet. For a heuristic function you could use a function that compared the desired url to the possible links. You could look for words that are similar. First you would probably want to look at the domain name and if you come across any links that are on that domain you would want to follow it. The actions would be to check all of the current link options for similar words or domain, and follow the most likely one. This would not necessarily be admissible, but would be your best bet. For websites, the url might not have anything to do with the content, which makes this particular activity hard. You might have better luck comparing the content of the site and looking for keywords in the content instead of in the link url.

function processPage(pageURL){

if (pageURL == goalURL){

println(‘BOOM! Winner!’);

}

for (links on page){

heuristic(link)

for (orderedLink in orderedLinks)}

//create a sorted list from lowest potential to highest potential

if (orderedLink.potential > heuristic(link).potential){

orderedlinks.add(I, heuristic(link))

}

}

}

return processLinks(orderedLinks)

}

function processLinks(sortedLinkArray){

for (var i=sortedLinkArray.length; i>0;i--){

var result = processPage(sortedLinkArray.get(I))

if (result){

return result;

}

}

return null;s

}