

CS4725/CS6705 - Fall 2017
Lab Activity # 1

Name: _____

Student ID: _____

Note: Your answers to these questions are to be handed in by 4:00 pm on **Wednesday, September 27, 2017**, but you are encouraged to complete these pages as soon as possible.

In this lab activity, we looked at the steps required to solve search problems using some provided Java code based on the algorithms in our textbook. The questions below are designed to get you to think about the different search methods that you are using.

The search problem being studied in this activity is the “Reverse” problem described during the lab session.

Here is a reminder of the steps that we followed in order for you to copy the provided code into your own account and to run the code:

- Create a new directory for yourself, change to that directory, create the required subdirectories under that directory and change to the **reverse** sub-sub-subdirectory.

```
mkdir AILabs
cd AILabs
mkdir -p aima/search/reverse
cd aima/search/reverse
```

- Copy the provided code into your **reverse** directory.

```
cp /fcs/courses/cs4725/F17/lab1/* . (Note the space and period after the *)
```

- Move back up to your **AILabs** directory, tell the system where to find a lot of the classes that have been provided to us, and then run the code.

```
cd ../../..
setenv CLASSPATH ./fcs/courses/cs4725/aima-java/build
java aima/search/reverse/ReverseDemo
```

1. Test results from “Reverse” test cases**Test case 1:** 4 5 6 7 3 2 1 8

Search method	Sol. found? (Y/N)	Nodes expanded	Solution found (sequence of actions)	# of actions in solution
BFS	Y	27	<u>4</u> , <u>7</u> , __, __, __, __, __, __, __	2
DLS, limit = 6			__, __, __, __, __, __, __, __, __	
DLS, limit = 5			__, __, __, __, __, __, __, __, __	
DLS, limit = 4			__, __, __, __, __, __, __, __, __	
DLS, limit = 3			__, __, __, __, __, __, __, __, __	
DLS, limit = 2			__, __, __, __, __, __, __, __, __	
IDS			__, __, __, __, __, __, __, __, __	

Test case 2: 6 7 8 3 5 4 2 1

Search method	Sol. found? (Y/N)	Nodes expanded	Solution found (sequence of actions)	# of actions in solution
BFS			__, __, __, __, __, __, __, __, __	
DLS, limit = 8			__, __, __, __, __, __, __, __, __	
DLS, limit = 6			__, __, __, __, __, __, __, __, __	
DLS, limit = 4			__, __, __, __, __, __, __, __, __	
DLS, limit = 3			__, __, __, __, __, __, __, __, __	
IDS			__, __, __, __, __, __, __, __, __	

Test case 3: 8 7 1 5 4 3 6 2

Search method	Sol. found? (Y/N)	Nodes expanded	Solution found (sequence of actions)	# of actions in solution
BFS			____, ____, ____, ____, ____, ____, ____, ____	
DLS, limit = 7			____, ____, ____, ____, ____, ____, ____, ____	
DLS, limit = 6			____, ____, ____, ____, ____, ____, ____, ____	
DLS, limit = 5			____, ____, ____, ____, ____, ____, ____, ____	
DLS, limit = 4			____, ____, ____, ____, ____, ____, ____, ____	
IDS			____, ____, ____, ____, ____, ____, ____, ____	

Test case 4: 3 7 1 4 6 2 5 8

Search method	Sol. found? (Y/N)	Nodes expanded	Solution found (sequence of actions)	# of actions in solution
BFS			____, ____, ____, ____, ____, ____, ____, ____	
DLS, limit = 8			____, ____, ____, ____, ____, ____, ____, ____	
DLS, limit = 7			____, ____, ____, ____, ____, ____, ____, ____	
DLS, limit = 6			____, ____, ____, ____, ____, ____, ____, ____	
IDS			____, ____, ____, ____, ____, ____, ____, ____	

2. Discussion questions - Uninformed search

(a) In class, we talked about the **optimality** of different search methods. Comment briefly on how your results on these specific test cases demonstrate what we learned.

For example, if one of the techniques is not optimal, point to specific test cases that demonstrate this for our example problem. If a technique **is** optimal, you don't have to prove it, but you should argue why your test results seem to support this.

(b) In class, we talked about the **completeness** of different search methods. Comment briefly on how your results on the specific test cases demonstrate what we learned.

(c) In class, we talked about the **time complexity** (number of nodes expanded) of different search methods. Comment briefly on the number of nodes expanded by the different techniques, relative to each other, in our test cases.

(d) If we are using tree search (without avoiding repeated nodes) and using the way in which we have ordered the successors of each node, standard **depth-first search** would be doomed to fail on this particular problem. Provide an example to demonstrate why this is the case.