CSCI 5430 Artificial Intelligence Assignment 4

Due 11:59pm, Nov 6th, Sunday, 2016

1. Explain how the graph search for sensorless problems can be revised to make use of the fact that any action sequence which is a solution for a belief state b is also a solution for a subset of b. (10 points)
2. In the slippery vacuum world, suppose the current belief state of the agent is {1,3} (see slides or textbook for definitions of the physical states 1~8). After the agent performs a Right action, an observation of [B, dirty] is sensed. What is the agent’s updated belief state? (10 points)
3. Implement a local search algorithm to solve the maximum independent set problem. See <https://en.wikipedia.org/wiki/Independent_set_(graph_theory)> for problem description. Choose from Hill-climbing search, Simulated annealing search, Local beam search, Genetic algorithms or one of their variants described in slides or textbook or other reference you found. You may choose to use Python, Java, or C/C++ to implement your algorithm. Input graph format is as follows: The first line specifies the number of vertices and edges; the following lines specifies all the edges, one line for each edge. Each edge is specified by its two end point labels. For a graph with n vertices and m edges, the valid vertex labels are from 0 to n-1 and the graph file will consist of m+1 lines. See example graph file gsample.txt. Output (to stdout) the independent set your algorithm found in the first line and its size in the second line. See the following sample output. Try to make your codes compilable and runnable on each of the major platforms Windows, Linux, Mac. (I will test your codes on two graph instances on Linux.) Your documentation need to specify how to compile and run your codes; specify what algorithm you have chosen to implement; what problem formulation and data structures you have chosen and how they fit into your implementation. About one page of explanation is expected unless you feel lesser amount is adequate. You will get 0 points if your code does not implement one of the requested algorithms. At least 40 points will be deducted if your code 1) does not compile, or 2) does not finish within 5 seconds, or 3) does not give correct results. (40 points (code)+ 40 points (documentation)=80 points)

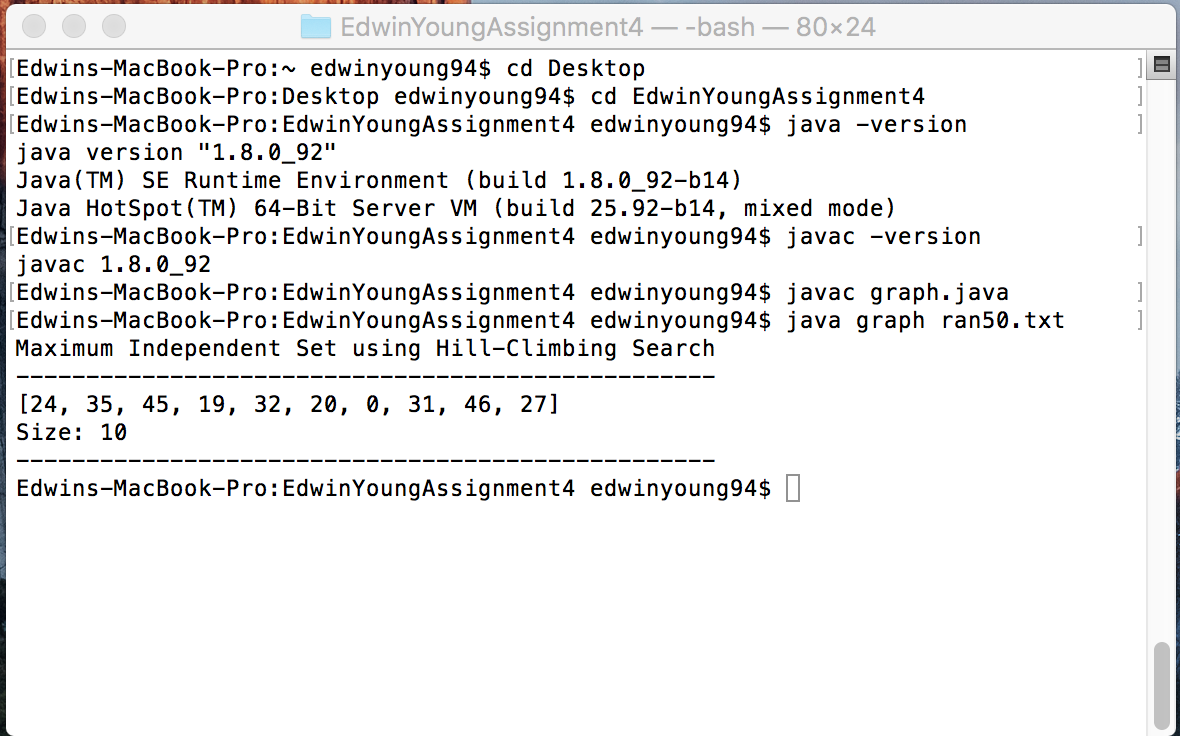
Sample output (not for gsample.txt):

1 2 6 9 15 20

Size: 6

**MAC OSX**

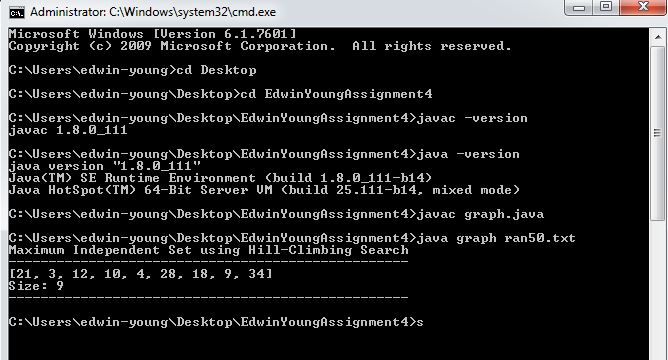
1. Download the .zip folder from Folio to Desktop
2. Extract to a folder to Desktop
3. Open Terminal and navigate to output folder on desktop
   1. For me it was
      1. cd Desktop
      2. cd EdwinYoungAssignment4
4. Make sure you have java and java complier installed by running the following commands(both should be installed before proceeding:
   1. java –version
   2. javac –version
5. Compile graph.java by typing following command into the terminal window
   1. javac graph.java
6. Run new java class that was just created from previous command by running the following command in the terminal window and add the .txt file you want to use for the vertices and edges(I used ran50.txt which is included in the .zip)
   1. java graph ran50.txt
7. Here is a screenshot of the terminal when I follow these steps:



**Windows**

**Completed this using Windows 7 Service Pack 1**

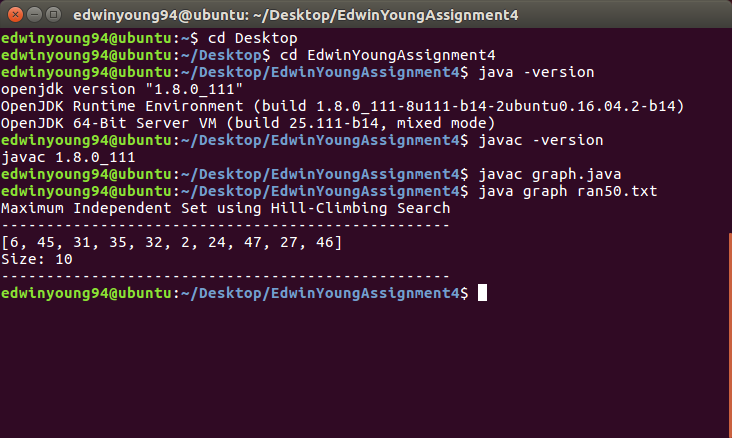
1. Download the .zip folder from Folio to Desktop
2. Extract to a folder to Desktop in a folder called “EdwinYoungAssignment4” or what ever is easiest for you to remember.
3. Open command prompt but hitting the windows key and typing “cmd” in start menu search bar then it “Enter” key and navigate to output folder on desktop
   1. For me it was
      1. cd Desktop
      2. cd EdwinYoungAssignment4
4. Make sure you have java and java complier installed by running the following commands(both should be installed before proceeding:
   1. java –version
   2. javac –version
5. Compile graph.java by typing following command into the command prompt window
   1. javac graph.java
6. Run new java class that was just created from previous command by running the following command in the command prompt window and add the .txt file you want to use for the vertices and edges(I used ran50.txt which is included in the .zip)
   1. java graph ran50.txt
7. Here is a screenshot of the command prompt when I follow these steps:



**Unbuntu**

**Completed using Ubuntu Desktop 16.04**

1. Download the .zip folder from Folio to Desktop
2. Extract to a folder on Desktop by clicking on it and naming folder for output “EdwinYoungAssignment4” or something that is easy to remember.
3. Open Terminal and navigate to output folder on desktop
   1. For me it was
      1. cd Desktop
      2. cd EdwinYoungAssignment4
4. Make sure you have java and java complier installed by running the following commands(both should be installed before proceeding:
   1. java –version
   2. javac –version
5. Compile graph.java by typing following command into the terminal window
   1. javac graph.java
6. Run new java class that was just created from previous command by running the following command in the terminal window and add the .txt file you want to use for the vertices and edges(I used ran50.txt which is included in the .zip)
   1. java graph ran50.txt
7. Here is a screenshot of the terminal when I follow these steps



**CentOS**

**Using version CentOS 7 DVD ISO**

1. Download the .zip folder from Folio to Desktop
2. Extract to a folder on Desktop by clicking on it and naming folder for output “EdwinYoungAssignment4” or something that is easy to remember.
3. Double click the output folder created on the desktop and right click in the window not on a file and select the option “Open in terminal
4. Make sure you have java and java complier installed by running the following commands(both should be installed before proceeding:
   1. java –version
   2. javac -version
5. Compile graph.java by typing following command into the terminal window
   1. Javac graph.java
6. Run new java class that was just created from previous command by running the following command in the terminal window and add the .txt file you want to use for the vertices and edges(I used ran50.txt which is included in the .zip)
   1. java graph ran50.txt
7. Here is a screenshot of the terminal when I follow these steps:

