Course: CSC520, Introduction to Artificial Intelligence
Homework 2
Student: Xusheng Xiao
Unity ID: xxiao2
Email: xxiao2@ncsu.edu

---

1. **(34 points) Consider the following English sentences.:**

   (a) John is a lawyer.

   (b) Lawyers are rich.

   (c) Rich people have big houses.

   (d) Big houses are a lot of work to maintain.

   (e) John has a house.

   Answer the following questions:

   (a) (10 points) Convert the sentences into first order predicate logic. Use the following lexicon:
   Functions: $houseof(X)$ – the house of X
   Predicates: $lawyer(X)$ – X is a lawyer
   $rich(X)$ – X is rich
   $big(X)$ – X is big
   $lotofwork(X)$ – X is a lot of work
   $house(X, Y)$ – the owner of house X is Y

      i. $lawyer(john)$
      ii. $\forall X(lawyer(X) \Rightarrow rich(X))$
      iii. $\forall X(rich(X) \Rightarrow house(houseof(X)) \wedge big(houseof(X)))$
      iv. $\forall X(big(houseof(X)) \Rightarrow lotofwork(houseof(X)))$
      v. $house(houseof(john))$

   (b) (12 points) Convert the logic statements into CNF.
      i. $lawyer(john)$
      ii. $\neg lawyer(X1) \vee rich(X1)$
      iii. $\neg rich(X2) \vee house(houseof(X2))$
      iv. $\neg rich(X3) \vee big(houseof(X3))$
      v. $\neg big(houseof(X4)) \vee lotofwork(houseof(X4))$
      vi. $house(houseof(john))$

   (c) (12 points) Using resolution, prove that "John's house is a lot of work to maintain."
   Conclusion: $house(houseof(john)) \wedge lotofwork(houseof(john))$

Negated conclusion:
1. $\neg house(houseof(john)) \vee \neg lotofwork(houseof(john))$

2.	$rich(john)$	$i + ii\{john/X\}$
3.	$house(houseof(john))$	$2 + iii\{john/X2\}$
4.	$big(houseof(john))$	$2 + iv\{john/X2\}$
5.	$lotofwork(houseof(john))$	$4 + v\{john/4\}$
6.	$\neg house(houseof(john))$	$5 + 1$
7.	$\emptyset$	$3 + 6$

2. **(24 points) Consider the following English sentences.**

- The College of Engineering is in Daniels Hall and the College of Business is in Nelson Hall.
- Computer Science and Civil Engineering are departments in the College of Engineering.
- Finance is a department in the College of Business.
- Smith is a professor in Computer Science, Jones in Civil Engineering, and Edison in Finance.

Now answer the following questions. Submit the final prolog code.

(a) (8 points) Convert the English sentences into a Prolog database.

```
in(ce,daniel).
in(cm,nelson).
department(cs,ce).
department(civil,ce).
department(finance,cm).
prof(smith,cs).
prof(jones,civil).
prof(edison,finance).
```

(b) (4 points) Write Prolog rules for each of the following statements and add the rules into the Prolog database.

- A faculty member of a department is also a faculty member in the corresponding College.

  ```
  prof(F,S) :- department(D,S),prof(F,D).
  ```

- The location of the department in the same as the location of corresponding College.

  ```
  in(D,L) :- department(D,C),in(C,L).
  ```

(c) (4 points) Now, write a query to find the location of Prof. Smith. and show the result. (Hint: the professor will be in same building as his/her department)

prof(smith,D),in(D,L).

(d) (4 points) Add to the Prolog database a rule with head location(X, Y) that will display the location of a faculty given a query such as location(smith,Where).

location(F,L) :- prof(F,D),in(D,L).

(e) (4 points) Now write a single query for each of the following questions and show the results.

- Name the faculty in the College of Engineering.

  prof(Faculty, ce)
- List ALL the occupants of Daniels Hall.

  in(Occupant, daniel)

3. **(12 points) Question 3.15 a, b, c, and d from Russell & Norvig, p. 116.**

4. **(30 points) Here is a road map of Romania map. (The numbers on the edges indicate the distance between the cities connected, but you don't distances them until Homework 3.).**

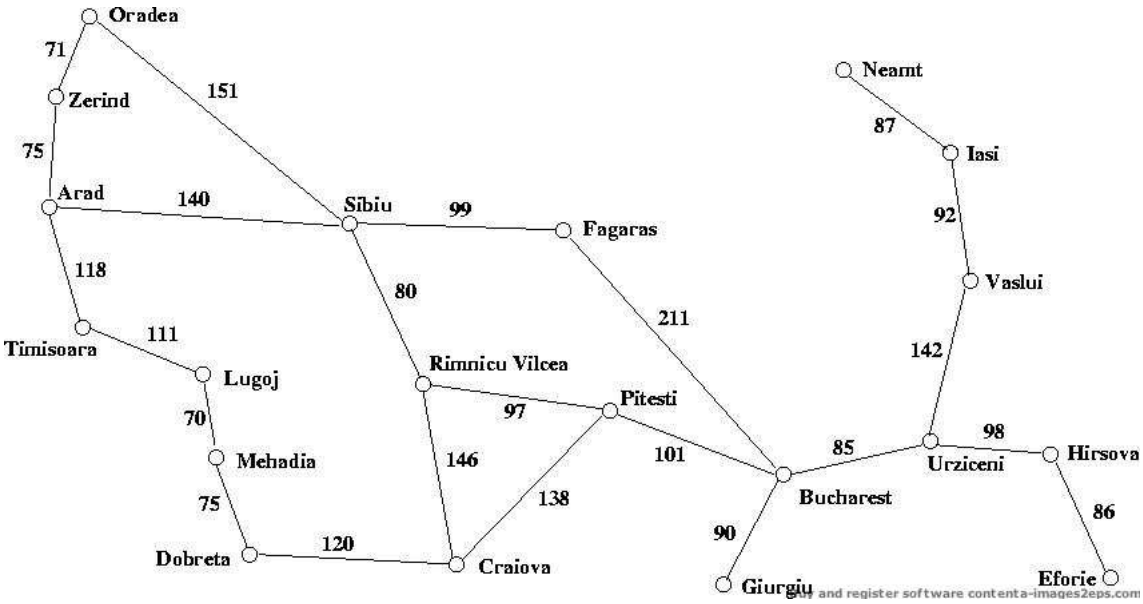
   For this assignment, you may use the Prolog source in the webnotes for dfs.pl, bfs2.pl and roads.pl.

   (a) (8 points) Consider the path from Arad to Bucharest and the path from Bucharest to Arad. Show the paths returned by DFS and BFS results for each case. Does either DFS or BFS return a path through the same set of cities for either? Give a reason for this behavior.

   (b) (6 points) Is there a case where Depth-First performs worse than Breadth-First (in terms of number of cities visited in the path, not the distance) ? If yes, what is it? If not, explain why.

   (c) (6 points) Is there a case where Breadth-First performs worse than Depth-First (in terms of number of cities visited in the path, not the distance)? If yes, what is it? If not, explain why.

   (d) (10 points) For the same graph, perform a hand-execution of Depth-First Iterative Deepening (DFID) with increment and cutoff initialized to 1, starting at Oradea. List the nodes in the order expanded and the state of the datastructure for the first five iterations of DFID. Expand the nodes alphabetically and insert them in nondecreasing



alphabetical order. Compare this list with Breadth-First Search. (No Prolog is required for the DFID portion of this question.).