CSE 463/563 Project # 3A (due on 11/17/14)

This assignment is about the use of negation in Prolog. In order to complete it, you have to install XSB on your machine (or use a CSE server), and create a text-file name "`Project3A.P`"; notice that the extension *must* be a capital P. This file will contain your Prolog code; as usual, you will submit it executing the command `submit_cse563` (or `submit_cse463`) on any CSE server. This is **individual work**. Identical solutions will be considered potential violations of academic integrity. At the beginning of `Project3A.P` please insert three Prolog comments, stating your name, your person number, and the phrase **"The submitted solutions are my individual work."**. Also, you should write the command for enabling auto-tabling (that is `:- auto_table.`). For example:

```
% <Your name>
% <Your UB person number>
% The submitted solutions are my individual work.
:- auto_table.
```

In this assignment you are allowed to declare only facts and clauses. You can use non-Horn clauses (hence, you can use negation). *Do not* use any other Prolog built-in predicate or operator, lists, or any arithmetic operator. Do not use external libraries.

Your goal is to analyze a given directed graph, and determine whether some properties hold or not. You can assume graphs are represented with the predicate `edge(G,A,B)`, stating that in the graph `G` there is a *directed* edge from node `A` to node `B`. For example, two directed graphs (named `italy` and `ny`) could be expressed as follows:

```
edge(italy, rome, milan).
edge(italy, milan, venice).
edge(italy, venice, rome).
edge(italy, palermo, syracuse).

edge(ny, nyc, albany).
edge(ny, nyc, rochester).
edge(ny, nyc, buffalo).
edge(ny, albany, rochester).
edge(ny, rochester,buffalo).
```

Step 1 (1 point)

Define a predicate `node(G,X)`, that is satisfied when `X` is a node in the graph `G`, that is `X` is either the source or the destination of an edge in `G`. You can assume there are no singleton nodes.

```
?- node(italy, rome) -> yes.
?- node(italy, nyc)  -> no.
```

Step 2 (2 points)

Define a predicate `tc(G,A,B)`, that is satisfied when the edge (`A`,`B`) belongs to the transitive closure of graph `G` (that is, when there is a directed path from `A` to `B`). For example:

```
?- tc(ny, albany, buffalo) -> yes.
?- tc(ny, buffalo, albany) -> no.
```

Step 3 (3 points)

Define a predicate `acyclic(G)`, that is satisfied when the graph `G` does not contain any directed cycle. For example:

```
?- acyclic(ny)    -> yes.
?- acyclic(italy) -> no.
```

Step 4 (3 points)

Define a predicate `connected(G)`, that is satisfied when there is an undirected path between each pair of nodes in graph `G`. For example:

```
?- connected(ny)    -> yes.
?- connected(italy) -> no.
```

Step 5 (3 points)

Assuming that graph `G` is acyclic, define a predicate `tr(G,A,B)` that is satisfied when the edge (`A`,`B`) belongs to the transitive reduction of graph `G`. The transitive reduction of a directed graph `G` is the subgraph of `G` with the fewest possible edges that has the same reachability relation as `G`.

```
?- tr(ny, nyc, buffalo)       -> no.
?- tr(ny, rochester, buffalo) -> yes.
```

HINT: when `G` is a directed acyclic graph a transitive reduction exists and is unique, the edge (`A`,`B`) belongs to it iff there is no other (longer) path from `A` to `B`.

Final notes

In order to define the above predicates you will need to declare some additional, accessory predicates. It is up to you to work out the details. As usual, it is highly recommended to test your solution w.r.t. non trivial graphs before submitting it.

CSE 463/563 Project # 3B (due on 11/17/14)

This assignment is about answer set programming. In order to complete it, you have to install DLV (`http://www.dlvsystem.com`) on your machine or use one of the CSE servers, and create a text-file named "`Project3B.dlv`". This file will contain your DLV code; you will submit it executing the command `submit_cse563` (or `submit_cse463`) on any CSE server. This is **individual work**. Identical solutions will be considered potential violations of academic integrity. At the beginning of `Project3B.dlv` please insert three comments, stating your name, your person number, and the phrase **"The submitted solutions are my individual work."**. For example:

```
% <Your name>
% <Your UB person number>
% The submitted solutions are my individual work.
```

In this assignment you are allowed to declare only facts and clauses.

Problem (8 points)

In Basin City there are all kinds of criminals. Your goal is to identify all criminals that are *gangsters*. A gangster is a criminal that is respected by all the other criminals and that shows respect for no one except, possibly, for other gangsters. According to the Basin City Police Department there is at least one gangster in the city and the following facts are true:

```
respects(marsellus_wallace, mark_gor_lee).
respects(mark_gor_lee, carlito_brigante).
respects(walter_kurtz, carlito_brigante).
respects(anton_chigurh, mark_gor_lee).
respects(charles_bronson, carlito_brigante).
respects(carlito_brigante, tony_montana).
respects(marsellus_wallace, tony_montana).
respects(tony_montana, vito_corleone).
respects(vito_corleone, carlito_brigante).
respects(anton_chigurh, vito_corleone).
respects(marsellus_wallace, carlito_brigante).
respects(tony_montana, carlito_brigante).
respects(anton_chigurh, carlito_brigante).
respects(marsellus_wallace, charles_bronson).
respects(charles_bronson, mark_gor_lee).
respects(charles_bronson, marsellus_wallace).
respects(walter_kurtz, vito_corleone).
respects(charles_bronson, tony_montana).
respects(marsellus_wallace, vito_corleone).
respects(charles_bronson, walter_kurtz).
respects(vito_corleone, tony_montana).
respects(mark_gor_lee, charles_bronson).
respects(carlito_brigante, vito_corleone).
respects(charles_bronson, vito_corleone).
respects(mark_gor_lee, marsellus_wallace).
respects(anton_chigurh, tony_montana).
respects(walter_kurtz, charles_bronson).
respects(walter_kurtz, tony_montana).
respects(charles_bronson, anton_chigurh).
respects(anton_chigurh, charles_bronson).
respects(walter_kurtz, marsellus_wallace).
respects(mark_gor_lee, vito_corleone).
respects(mark_gor_lee, tony_montana).
respects(anton_chigurh, marsellus_wallace).
respects(walter_kurtz, mark_gor_lee).
```

No other ground atom of the predicate `respect` is true. Use DLV for identifying all the gangsters in Basin City. Make sure that the invocation `dlv -pfilter=gangster Project3B.dlv` returns the list of gangsters. The output should look like this:

```
{gangster(huey), gangster(louie), gangster(dewey)}
```