Assignment 2

Problem 1

(c)

% descendant(X,Y) means X is a descendant of Y.

descendant(X, Y) :- ancestor(Y, X).

% sibling(X,Y) means X and Y share a parent P.

sibling(X, Y) :- parent(P, X), parent(P, Y), X != Y.

% If there is no distinction between sibling(X1,X2) and sibling(X2,X1), then we can use X<Y instead.

```
C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>dlv.mingw family.dlv -filter=descendant
DLV [build BEN/Dec 17 2012   gcc 4.6.1]

{descendant(john,william), descendant(james,william), descendant(james,john), descendant(bill,william), descendant(bill,john), descendant(bill,james),
 descendant(bill,sue), descendant(carol,william), descendant(carol,john), descendant(carol,james), descendant(carol,sue)}

C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>
```

```
C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>dlv.mingw family.dlv -filter=sibling
DLV [build BEN/Dec 17 2012   gcc 4.6.1]

{sibling(bill,carol), sibling(carol,bill)}
```

(d)

% ICs (Integrity Constraints) -- RULES to find "bad" (inconsistent) data

% Warm-up

% Assume the IC says: "every person must have a parent".

% How can we guarantee that?

%

% First we find persons who do NOT violate the constraint,

% that is, we find persons who have parents:

%

has_parent(X) :- parent(_, X).

% person(X) means X is a parent or a child.

person(X) :- parent(X,_).

```
person(X) :- parent(_,X).

% Now we can easily find persons who violate the has_parent constraint.
% We can use a convention, e.g., "icv_NNN" to mark [i]ntegrity [c]onstraint [v]iolations
%
icv_no_parent(X) :- person(X), not has_parent(X).
% Mom & Dad
father(X, Y) :-
        parent(X, Y), male(X).
mother(X, Y) :-
        parent(X, Y), female(X).
% Every person has a father and a mother.
has_mom_and_dad(X) :- mother(M, X), father(F, X), M != F.

% icv_no_mom_or_dad(X) means any person who does not have a mom or dad.
icv_no_mom_or_dad(X) :- person(X), not has_mom_and_dad(X).
```

```
C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>dlv.mingw family.dlv -filter=icv_no_parent
DLV [build BEN/Dec 17 2012   gcc 4.6.1]

{icv_no_parent(william), icv_no_parent(sue)}
```

```
C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>dlv.mingw family.dlv -filter=icv_no_mom_or_dad
DLV [build BEN/Dec 17 2012   gcc 4.6.1]

{icv_no_mom_or_dad(william), icv_no_mom_or_dad(john), icv_no_mom_or_dad(james), icv_no_mom_or_dad(sue)}
```

Problem 2

(a) (FD-1)

```
% Problem 2a FD-1: if a row agrees with another row on the key attribute PID,
% then it should agree on ALL other attributes.
% I suppose null value should also be reported.
```

icv_fd1(author_violation,X,Y1,Y2) :- publication(X,Y1,_,_,_,_,_,_,_,_), publication(X,Y2,_,_,_,_,_,_,_,_), Y1 < Y2.

icv_fd1(year_violation,X,Y1,Y2) :- publication(X,_,Y1,_,_,_,_,_,_,_), publication(X,_,Y2,_,_,_,_,_,_,_), Y1 < Y2.

icv_fd1(title_violation,X,Y1,Y2) :- publication(X,_,_,Y1,_,_,_,_,_,_), publication(X,_,_,Y2,_,_,_,_,_,_), Y1 < Y2.

icv_fd1(journal_violation,X,Y1,Y2) :- publication(X,_,_,_,Y1,_,_,_,_,_), publication(X,_,_,_,Y2,_,_,_,_,_), Y1 < Y2.

icv_fd1(vol_violation,X,Y1,Y2) :- publication(X,_,_,_,_,Y1,_,_,_,_), publication(X,_,_,_,_,Y2,_,_,_,_), Y1 < Y2.

icv_fd1(no_violation,X,Y1,Y2) :- publication(X,_,_,_,_,_,Y1,_,_,_), publication(X,_,_,_,_,_,Y2,_,_,_), Y1 < Y2.

icv_fd1(fp_violation,X,Y1,Y2) :- publication(X,_,_,_,_,_,_,Y1,_,_), publication(X,_,_,_,_,_,_,Y2,_,_), Y1 < Y2.

icv_fd1(lp_violation,X,Y1,Y2) :- publication(X,_,_,_,_,_,_,_,Y1,_), publication(X,_,_,_,_,_,_,_,Y2,_), Y1 < Y2.

icv_fd1(publisher_violation,X,Y1,Y2) :- publication(X,_,_,_,_,_,_,_,_,Y1), publication(X,_,_,_,_,_,_,_,_,Y2), Y1 < Y2.

```
C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>dlv.mingw hw2-problem-2.dlv -filter=icv_fd1
DLV [build BEN/Dec 17 2012   gcc 4.6.1]

{icv_fd1(author_violation,4407,doe,kummel), icv_fd1(year_violation,4407,1969,2015), icv_fd1(title_violation
,4407,ammonoids,foobar), icv_fd1(vol_violation,4407,10,137), icv_fd1(no_violation,4407,1,3), icv_fd1(fp_vio
lation,4407,10,476), icv_fd1(lp_violation,4407,null,1), icv_fd1(publisher_violation,4407,null,publisher2)}
```

(a) (FD-2)

% Problem 2a FD-2: every journal has a single publisher

% I suppose null value should also be reported.

icv_fd2(J,P1,P2) :- publication(_, _, _, _, J, _, _, _, _, P1), publication(_, _, _, _, J, _, _, _, _, P2), P1 < P2.

```
C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>dlv.mingw hw2-problem-2.dlv -filter=icv_fd2
DLV [build BEN/Dec 17 2012   gcc 4.6.1]

{icv_fd2(bullmcz,publisher1,publisher2), icv_fd2(bullmcz,null,publisher1), icv_fd2(bullmcz,null,publisher2)}
```

(a) (NC-1)

% Problem 2a NC-1: The last page Lp cannot be smaller than the first page Fp

% I suppose null value for a page number is not a violation.

icv_nc1(I,F,L) :- publication(I, _, _, _, _, _, _, F, L, _), F != null, L != null, F > L.

```
C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>dlv.mingw hw2-problem-2.dlv -filter=icv_nc1
DLV [build BEN/Dec 17 2012   gcc 4.6.1]

{icv_nc1(4407,10,1), icv_nc1(6755,91,9)}
```

(b) (ID)

% Problem 2b ID (Inclusion Dependency):

% Every cited publication in CITES also occurs in PUBLICATION.

% Note: Publications P2 in the second column of cites(P1,P2) constitute all

% *cited* publications, so checking P1 isn't required!

% (If P1 is included, all *citing* and *cited* pubs are checked for inclusion

% in the Publication table. Wasn't required but is a useful check, too.)

% Auxiliary relation: Unary relation to collect just the pub-ids.

pub_id(I) :- publication(I,_,_,_,_,_,_,_,_,_).

% The CITES[P2] \subseteq PUBLICATION[Pid] is violated if there is a P2

% that's not among the pubs in PUBLICATION:

icv_id(cited,I) :- cites(_,I), not pub_id(I).

% If you want to check *citing* articles as well, use this rule:

icv_id(citing,I) :- cites(I,_), not pub_id(I).

```
C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>dlv.mingw hw2-problem-2.dlv -filter=icv_id
DLV [build BEN/Dec 17 2012   gcc 4.6.1]

{icv_id(cited,2020), icv_id(cited,3799), icv_id(citing,3799), icv_id(citing,4711)}
```

(b) (NC-2)

% Problem 2b NC-2: If P1 cites P2 then P2's year of publication

% cannot be greater than P1.

```
icv_nc2(I1,I2,Y1,Y2) :-
        cites(I1,I2),
        publication(I1,_,Y1,_,_,_,_,_,_),
        publication(I2,_,Y2,_,_,_,_,_,_),
        Y1 < Y2.
```

```
C:\Documents\GSLIS\590 Data Cleaning\Assignment 2>dlv.mingw hw2-problem-2.dlv -filter=icv_nc2
DLV [build BEN/Dec 17 2012   gcc 4.6.1]

{icv_nc2(2044,2580,1934,1962)}
```