Temporal Reasoning in CLIPS

The goal of this exercise is to program a system for temporal reasoning using rules written in CLIPS 6. Your system should use the following templates for RELATIONS and TEMPORARY. The table of transitivity is defined by the deffacts named TABLE-OF-TRANSITIVITY listed below. You will use the functions UNION and INTERSECTION listed below.

In the list of temporal relations, the symbols ">", "<", and "=" correspond to CLIPS keywords. These must be replaced with "b" (before), "bi" (before-inverse) and "e" (equals).

```
(deftemplate RELATIONS (slot int1) (slot int2) (multislot relations))
(deftemplate TEMPORARY (slot name) (slot int1) (slot int2) (multislot
relations))

(deftemplate TABLE  (slot r1)  (slot r2) (multislot relations) )
(deffacts TABLE-OF-TRANSITIVITY
      (TABLE (r1 b) (r2 b) (relations b))
      (TABLE (r1 b) (r2 bi) (relations b bi d di o oi m mi s si f fi e))
      (TABLE (r1 b) (r2 d) (relations b o m d s))
      (TABLE (r1 b) (r2 di) (relations b))
      (TABLE (r1 b) (r2 o) (relations b))
      (TABLE (r1 b) (r2 oi) (relations b o m d s))
      (TABLE (r1 b) (r2 m) (relations b))
      (TABLE (r1 b) (r2 mi) (relations b o m d s))
      (TABLE (r1 b) (r2 s) (relations b))
      (TABLE (r1 b) (r2 si) (relations b))
      (TABLE (r1 b) (r2 f) (relations b o m d s))
      (TABLE (r1 b) (r2 fi) (relations b))

.....
      (TABLE (r1 fi) (r2 fi) (relations fi))
)

(deffunction UNION (?r1 ?r2)
   (progn$ (?e ?r1)
      (if (not (member ?e ?r2)) then (bind ?r2 (insert$ ?r2 1 ?e))))
   (return ?r2)
)

(deffunction INTERSECTION (?r1 ?r2)
   (bind ?r3 (create$))
   (progn$ (?e ?r1)
      (if (member ?e ?r2) then (bind ?r3 (insert$ ?r3 1 ?e)) ))
   (return ?r3)
)
```

Suppose that you have three intervals, ?A, ?B, and ?C as well as a list of possible relations ?RAB from ?A to ?B, and a list of possible relations ?RBC from ?B to ?C. The generation of the list of relations from ?A to ?C will be performed by three rules named R1, R2 and R3.

a) (2 points) For each pair of relations (?rab, ?rbc) such that ?rab ∈ ?RAB and ?rbc ∈ ?RBC, the rule R1 will generate a fact of type TEMPORARY using the table of transitivity. Write the rule R1.

b) (2 points) Whenever there exists two facts of type TEMPORARY with the same value for int1 and int2, but with different names, the lists of relations given by these facts must be combined using the function UNION. Write the rule R2 that deletes the two facts and creates a new fact of type TEMPORARY, in which the list of possible relations is the UNION of the lists from the two facts.

c) (2 points) When all of the facts of type TEMPORARY have been combined, the resulting list should be used to create a fact of type RELATIONS. Write the rule R3 that creates this new fact. This same rule should delete the fact of type TEMPORARY. Be careful that all of the TEMPORARY lists have been fused before this rule fires.

d) (2 points) Write the rule R3 that detects when two lists of relations have been generated for the same pair of intervals. This rule should delete the two facts of type RELATIONS and create a new fact with the intersection of the possible set of temporal relations.