This assignment is from the Third Edition of Bratko, Chapter 4. You may access the sections you need for this assignment here.

`http://www.cslab.pepperdine.edu/warford/cosc450/Bratko-ch4-3rd-ed.pdf`

1. Study Bratko, Third Edition, Sections 4.1 to 4.3.

   For the programming part of the assignment download this text file

   `http://www.cslab.pepperdine.edu/warford/cosc450/a11.txt`

   which you can modify and hand in. You will need to change the file extension from `a11.txt` to `a11.pl`.

2. Do Bratko, Third Edition, Exercise 4.1.
   Hand in your solution as `/* */` style comments in `a11.pl`. Put your query on a separate line in the comment section as follows:

   ```
   /*
   query4.1(a), families without children
   Put your query here.

   query4.1(b), all employed children
   Put your query here.

   query4.1(c), employed wives and unemployed husbands
   Put your query here.

   query4.1(d), children whose parents differ in age by at least 15 years
   Put your query here.
   */
   ```

   so I can copy and paste your queries to test them. Write queries, not rules.

   The `a11.pl` file has the facts of the family database and has been augmented with families for which the other queries can be tested. For example, 4.1(b) is a query for employed children, so families have been added with at least one child who is employed. Be sure to include whatever utilities you need for your queries to work.

   Here are some hints. For 4.1(a) and (b), the query is a single goal.

   For 4.1(d), first match `family` to instantiate variables `Husband`, `Wife`, and `Children`. Then, use `dateofbirth` twice to instantiate `HusbandYOB` and `WifeYOB`. Here is the goal that will succeed if their ages differ by 15 years or more.

   ```
   (HusbandYOB - WifeYOB >= 15
   ;
   WifeYOB - HusbandYOB >= 15)
   ```

3. Do Bratko, Third Edition, Exercise 4.2.
   The query

```
?- twins( C1, C2)
```

should list all the twins. First, you should match `family` to get an instantiation of `Children`. Then, use the `del/3` predicate to get a list of children without `Child1`. `Child2` is a twin if he is a member of the smaller list and his date of birth is the same as that of `Child1`. You can use the builtin `member/2` predicate, but you should include our definition of `del/3` in your solution.

4. Do Bratko, Third Edition, Exercise 4.3.
   Here is an example query.

```
?- nth_member( 2, [a,b,c,d,e], X).

X = b
```

5. Do Problem Nondeterministic FSM.
   Write a Prolog program that implements the nondeterministic finite state machine (FSM) for integers according to Warford, Figure 7.17.

   http://www.cslab.pepperdine.edu/warford/cosc450/warford-fig-7-17.pdf

   The input alphabet should be the atoms `plus`, `minus`, `digit`. The states should be `si`, `sf`, and `sm` corresponding to I, F, and M in the figure. For example, the queries

```
accepts( si, [plus, digit]).
```

and

```
accepts( si, [digit, digit]).
```

should succeed and the query

```
accepts( si, [digit, plus, digit]).
```

should fail. Note that `a11.pl` contains the `accepts/2` predicates for an arbitrary finite state machine, so you only need to write the facts for this specific machine.