CS 435/535 Assignment #8 – Spring 2021

Project Overview: In this project, you will use **Prolog** to complete a simple project related to three kinds of employees: `salaried` employees, `hourly` employees, and `commission` employees. A salaried employee has three attributes: first name, last name, and weekly salary. An hourly employee has four attributes: first name, last name, the number of hours worked in a week and the hourly rate. A commission employee has five attributes: first name, last name, the minimal salary, the amount of sales, and the commission rate. The following shows a list of employees and the format used in the list.

```
salaried Jeremy Greer 170
salaried Jane Brown 5000
salaried George Miller 1900
salaried Robert Johnson 3000
hourly Maria Garcia 20 8.50
hourly Carlton West 42 12.50
hourly Viola Jennings 60 17.50
commission Mary Smith 300 19000 0.1
commission Floyd Jenkins 500 3000 0.12
commission Rose Harvey 500 20000 0.15
```

```
salaried <first> <last> <salary>
hourly <first> <last> <hours> <rate>
commission <first> <last> <min_salary> <amount> <rate>
```

Your first job is to create a Prolog file named `facts.pl` in which you will use predicates to represent all the employees listed above. You need to write three predicates (`salaried`, `hourly`, and `commission)` to describe three kinds of employees and use the parameter order that is the same as the order in the above format. For this project, you can assume the list of employees will be fixed.

Your second job is to create another Prolog file named `rules.pl` in which you will write the rules to compute the pay for each kind of employees. The `pay` predicate shall be of the following format: `pay(First, Last, Pay)`. The pay for a particular employee is computed as follows.

- For a salaried employee, the pay is simply the weekly salary.
- For an hourly employee, the pay is based on the number of hours worked in a week and the hour rate. However, the first 40 hours is paid exactly at the hourly rate, the next 10 hours (40 to 50) is paid at 1.5 times the hourly rate, and any hours over 50 is paid at 2.0 times the hourly rate.
- For a commission employee, if the commission computed (the sales amount times the commission rate) is less than the minimum salary, the pay is the minimum salary instead of the commission.

The facts and the rules form a database. You will create the third Prolog file named `functions.pl` in which you will write six predicates (they can be viewed as the rules too) that can be used to query the database. These six predicates are listed below. The `Op` could be `'=='`, `'!='`, `'>'`, `'>='`, `'<'` and `'<='`, and `Ref` is a reference value for `Pay`. `Op` and `Ref` are used to identify a sublist of employees that satisfy a condition of the `Pay Op Ref` format, `Pay '>=' 1200`, for example.

`list(Op, Ref, List). %` check whether the sublist is the same as `List.`
`count(Op, Ref, Count). %` Check whether `Count` is the number of employees in the sublist.
`min(Op, Ref, Min). %` Check whether `Min` is the minimum pay of the employees in the sublist.
`max(Op, Ref, Max). %` Check whether `Max` is the maximum pay of the employees in the sublist.
`total(Op, Ref, Total). %` Check whether `Total` is the total pay of the employees in the sublist.
`avg(Op, Ref, Avg). %` Check if `Avg` whether the average pay of the employees in the sublist.

A sample execution of the program at the end shows how to load the facts and the rules, and how to issue a query, assuming the database has 10 employees as listed above.

What You Need To Do

- Create a directory named `project8` for this assignment. Under `project8`, create three files: `facts.pl`, `rules.pl` and `functions.pl` as specified above. Make sure to use the exact names for these three files and for all the predicates defined in these three files.
- When you are ready to submit your project, compress your `project8` directory into a single (compressed) zip file, `project8.zip`.
- Once you have a compressed zip file named `project8.zip`, submit that zip file to Blackboard.
- Your submission will be graded on `cs-parallel`. Make sure to test it on that machine before submission.
- Make sure to follow the above instructions exactly. Otherwise we may not be able to grade your submission.

Assignment #8 is due at 11:59pm on Monday, April 16. Late projects are not accepted.

This document including its associated files is for your own personal use only.
You may not post this document or a portion of this document to a site
such as chegg.com without prior written authorization.

An assignment shall be completed individually, with no sharing of code or solutions.
All submissions will go through MOSS (Measure Of Software Similarity) for similarity check.
The University of Alabama's Code of Academic Conduct will be rigorously enforced.

A sample execution of the program

```
gprolog

| ?- [facts].

| ?- [rules].

| ?- [functions].

| ?- list('>=', 0, List).

List =
[['Jeremy','Greer',170],['Jane','Brown',5000],['George','Miller',1900],['R
obert','Johnson',3000],['Maria','Garcia',170.0],['Carlton','West',537.5],[
'Viola','Jennings',1312.5],['Mary','Smith',1900.0],['Rose','Harvey',3000.0
],['Floyd','Jenkins',500]]

| ?- count('>=', 0, Count).

Count = 10

| ?- max('<', 1900, Max).

Max = 1312.5

| ?- min('>', 1900, Min).

Min = 3000

| ?- list('==', 3000, List).

List = [['Robert','Johnson',3000],['Rose','Harvey',3000.0]]

| ?- total('>=', 700, Total).

Total = 16112.5

| ?- avg('<=', 5000, Avg).

Avg = 1749.0

| ?- halt.
```