CS_125
LOGIC PROGRAMMING
(Attempt 2 questions out of 3)

Question 1.

(a) Suppose a Prolog database defines predicates `mother/2` and `father/2` as a collection of facts.

  (i) Write down queries that correspond to the following questions:
    - Is Tom Mary's father?
    - Who is Tom's mother?

  (ii) Define the predicate `parent/2`.

  (iii) Define the predicate `descendant/2`.

  (iv) Define a predicate `all_descendants/2` that computes a list of all descendants of a person. You may use the built-in predicate `setof/3`.

[11 marks]

(b) A restaurant has the following pizzas on the menu (with name, toppings and price):

```
pizza(hawaiian,[cheese, ham, pineaple],4.50).
pizza(bambini,[cheese,tomato]),2.80.
pizza(napoli,[cheese,tomato,anchovis],3.30).
pizza(diavolo,[cheese,tomato,peperoni,chilli],3.90).
pizza(marinara,[cheese,tomato,mussels,tuna,calamari],4.50).
pizza(marco_polo,[cheese,tomato,ham,olives],4.20).
```

  (i) Write down queries that correspond to the following questions.
    - Which pizzas have at least three toppings and cost less than 4 pounds?
    - Which pizzas have ham but not pineapple?

  (ii) Write a program that computes the cost effectiveness of a pizza, that is, the price divided by the number of toppings.

You may use the built-in predicates `member/2`, `length/2` and `not/1`.

[8 marks]

(c) Which of the following terms are unifiable? In case the terms are unifiable, compute their most general unifier, otherwise explain why they are not unifiable.

  (i) $f(X, g(Z, Z), Z) \quad f(g(Y, Y), Y, c)$

  (ii) $f(X, g(Z, Z), X) \quad f(g(Y, Y), Y, c)$

[6 marks]

Question 2.

(a) Determine Prolog's answers to the following queries.

(i) `?- X = 1 + 1.`

(ii) `?- X is 1 + 1.`

(iii) `?- X is 1 + Y, Y is 5.`

(iv) `?- Y is 5, X is 1 + Y.`

(v) `?- Y is 5, X is 1 + Y, X < 3.`

(vi) `?- [1,2] = [X|T].`

(vii) `?- [X] = [X,X].`

(viii) `?- [X] = [X|X].`

[8 marks]

(b) Define a predicate `all_greater/2`, such that for a list of numbers `L` and a number `N` the atomic formula `all_greater(L,N)` means that all numbers occurring in `L` are greater than `N`. Use anonymous variables in all appropriate places.

[7 marks]

(c) Explain how Prolog solves a query by answering the following questions:

(i) What *proof calculus* does Prolog use? Briefly explain this calculus.

(ii) What is and what is the role of *unification*?

(iii) What does *backtracking* mean?

(iv) Which *search strategy* does Prolog apply? Briefly explain this strategy.

Your may back up your answers by drawing suitable commented illustrations.

[10 marks]

Question 3.

(a) (i) Explain the effect of the cut predicate. You may support your explanation by a suitable diagram.

(ii) Explain how the cut is used to define

- *if then else*;
- *negation as failure.*

[7 marks]

(b) Consider the following program:

```
p(X,[Y|_]) :- X = Y, !, X > 3.
p(X,[_|L]) :- p(X,L).
```

(i) Draw resolution trees to find Prolog's answers to the following queries.

```
?- p(X,[2,4]).
?- p(4,[2,4]).
```

(ii) Which of the queries in (i) would have been answered differently if the cut were removed?

(iii) Is the cut in the definition of the predicate `p` a green cut or a red cut?

[8 marks]

(c) (i) Define a predicate `sum_of_squares/2` that for a natural number n computes the sum of the square numbers up to n, that is,

$$\sum_{i \leq n} i^2 = 1 + 4 + 9 + \ldots + n^2.$$

Make sure that your program does not loop when '`;`' is pressed.

(ii) Use the predicate `sum_of_squares/2` to write a program that repeatedly asks the user to enter a natural number n and then writes the sum of the square numbers up to n to the standard output. The program should terminate when the user enters anything that is not a natural number.

[10 marks]