1A)

1. Prolog is suitable for knowledge representation due to the fact that it deals with logic allowing us to store facts and rules and use them. Prolog differs from normal computers as it does this is a direct way.
2. A) Rules in English are characterised by the presence of the word ‘if’.

B) Facts are stored in Prologs internal database.

C) When a variable is matched to a constant, the variable becomes instantiated to the constant.

D) Backtracking is when the interpreter attempts to solve a query in a different way because a sub-goal has failed

E) The process by which rules are selected by matching variables and constants is called unification.

G) Functions such as ‘fail’ and ‘write’ are usually referred to as built-in predicates.

3) tree(elm).

4) tree(elm). tree(birch). tree(oak). tree(X), write(X), nl, fail.

1B)

1. A) car(ford, Mod, Trim, Country, Cc, Type, Price), write(Mod),nl, write(Trim),nl, write(Country),nl, write(Cc),nl, write(Type),nl, write(Price),nl, fail.

B) car(ford, Mod, Trim, Origin, Capac, saloon, Price),

(Price < 9000),

(Capac > 1300).

C) car(rover, ModRover, \_, \_, \_, \_, PriceRover), car(fiat, ModFiat, \_, \_, \_, \_, PriceFiat), PriceRover > PriceFiat, write(ModRover),nl, fail.

1C)

1. Man = Rover then address= 18 Beadle Road, Cowley

**Address** = '21 Tinsgate, Dagenham',  
**Man** = ford,  
**Model** = fiesta

**Address** = '21 Tinsgate, Dagenham',  
**Man** = ford,  
**Model** = orion

**Address** = '21 Tinsgate, Dagenham',  
**Man** = ford,  
**Model** = orion

**Address** = '21 Tinsgate, Dagenham',  
**Man** = ford,  
**Model** = orion

**Address** = '333 Via Alphonse, Turin',  
**Man** = fiat,  
**Model** = uno

**Address** = '333 Via Alphonse, Turin',  
**Man** = fiat,  
**Model** = uno

**Address** = '18 Beadle Road, Cowley',  
**Man** = rover,  
**Model** = metro

**Address** = '18 Beadle Road, Cowley',  
**Man** = rover,  
**Model** = metro

2) car(Man, metro, mg, Country, \_, \_, \_), supplier(Man,Country,Address,\_).

3) car(Man, \_, \_, uk, \_, saloon, \_), supplier(Man,uk,\_,Tel).

4) car(Man, \_, \_, \_, Cc, \_, \_), Cc > 1300, supplier(Man,\_,\_,Tel).

Assignment= car(Man, \_, \_, \_, Cc, \_, \_), Cc > 1000, Cc =< 1600, supplier(Man,\_,Address,Tel).