

HW4

CS571

LOVELIN ANAND EDWARD PAUL

B00669954

1.

$\forall X, Y \text{ parent}(X, Y) \Rightarrow \text{ancestor}(X, Y).$

$\forall X, Y, Z \text{ parent}(X, Y) \vee \text{ancestor}(Y, Z) \Rightarrow \text{ancestor}(X, Z).$

2.

a) $f(x) :-$

$p(x),$
 $g1(x).$

$f(x) :-$

$q(x),$
 $g2(x).$

$f(x) :-$

$\backslash+ p(x),$
 $\backslash+ q(x),$
 $g3(x).$

b) $f(x) :-$

$p(x),$
 $g1(x),$
 $!.$

$f(x) :-$

$q(x),$
 $g2(x),$
 $!.$

$f(x) :-$

$\backslash+ p(x),$
 $\backslash+ q(x),$
 $g3(x).$

3.

`server() ->`

`spawn(module, loop, [0]). % Initializing the counter as zero.`

`loop(Counter) ->`

`receive`

`{ClientId, _ ...} ->`

`Counter = Counter + 1, % Updating the counter for each pattern.`

`% other actions`

`loop(Counter). % Passing the updated counter recursively.`

`{ClientPid, stop} ->`

`ClientPid ! {stopped, Counter},`

`true.`

end.

4.

We can avoid this latency problem by using pool of servers and a main supervisor to distribute the requests to these pool of servers based on availability or round robin method. But still this solution does not solve this problem completely in cases like if there are N workers and N “1 second” requests, all other requests will be stalled. Another solution can have just two workers, one to process “1 second” request, another to process “1 millisecond” request and a supervisor to distribute based on the request.

7.

```
class C {
    C f(C c) {
        System.out.println("1"); return new SC();    // 1
    }
}
class SC extends C {
    C f(SC sc) {
        System.out.println("2");                    // 2
        return new SC();
    }
    SC f(C c) {
        System.out.println("3");                    // 3
        return new SC();
    }
}

C c = new SC();
SC sc = new SC();

c.f(sc.f(sc));
sc.f(c.f(sc));
```

Output :

```
2 // printed while calling sc.f(sc) as the object reference points to SC.
3 // printed while calling c.f(new SC()) as the object reference points to C.
3 // printed while calling c.f(sc) as the object reference points to SC.
2 // printed while calling sc.f(new SC()) as object reference points to SC.
```

8.

a)

| | |
|----|----|
| | 12 |
| b) | 16 |
| | 16 |
| c) | 16 |
| | 14 |