1.

a) Stack: D T# Input: 1,0#
b) Stack: T# Input: ,0#
c) Stack: ',' L# Input: ,0#
d) Stack: L# Input: 0#

2.

- E -> O
- O -> O '==' A | A
- A -> A '!=' X | X
- X -> X '^' U | U
- U -> '!' U | P
- P -> '(' E ')' | 'x'

3.

a) Z -> '++' Z
b) C -> '(' E ')'
c) E -> A X
d) X -> epsilon

4.

a) (E/E)
b) ((E/E)*E)
c) ((x/E)*E)
d) ((x/2)*E)

5.

a) Yes
b) No
c) Yes
d) No

6.

a) Logic expressions

- (J V L) => C (Premise)
- ~T (Premise)
- C => T (Premise)
- ~J (Conclusion)

b) Clauses

- (~J V C) ^ (~L V C)
- ~T
- (~C V T)

c) Justification
Prove: ~J

- J, Assume the negation of the conclusion
- J and (J V L) => C resolves to L => C
- From ~T and C => T we have ~C
- But L => C and ~C contradict because L and ~L can't resolve
- So assumption of J is true must be incorrect, and ~J is proven

7.

a) Rules as clauses
- cousin(C1, C2) :- grandchild(C1, GP), grandchild(C2, GP).
- grandchild(C, GP) :- child(C, P), child(P, GP).

b) Proof

Premises:

- cousin(C1, C2) :- grandchild(C1, GP), grandchild(C2, GP).
- grandchild(C, GP) :- child(C, P), child(P, GP).
- child('Tim', 'Bea').
- child('Sue', 'Ned').
- child('Ned', 'Bea').
- child('Ann', 'Tim').

Query:

- cousin('Ann', 'Sue')?

Proof:

- cousin('Ann', 'Sue') :- grandchild('Ann', GP), grandchild('Sue', GP) (From premise 1)
- grandchild('Ann', GP) (Instantiate 1 with C1 = 'Ann')
- child('Ann', P), child(P, GP) (From premise 2)
- child('Ann', P), child(P, GP), child('Sue', GP) (From premise 4 and instantiation of 2 with GP = 'Ned')
- child('Ann', P), child(P, 'Ned'), child('Sue', 'Ned') (From premise 5 and instantiation of 4 with GP = 'Bea')
- child('Ann', 'Tim'), child('Tim', 'Bea'), child('Sue', 'Ned') (From premise 6 and instantiation of 5 with P = 'Tim').
- 'Tim' != 'Sue' (Contradiction from premise 6).
- Therefore, cousin('Ann', 'Sue') is false

8.

a) Inductive hypothesis
- Assume that P(n) is true for some integer n ≥ 1,
- 1*2*3 + 2*3*4 + ... + n(n+1)(n+2) = (n(n+1)(n+2)(n+3))/4

b) To be proven in the inductive step
- Prove that P(n+1) is true,
- 1*2*3 + 2*3*4 + ... + n(n+1)(n+2)(n+3) = (n(n+1)(n+2)(n+3)(n+4))/4

c) Result of using the inductive hypothesis
- P(n+1) = (n(n+1)(n+2)(n+3))/4 + (n+1)(n+2)(n+3)

d) Algebraic manipulation
- = P(n+1) = (n(n+1)(n+2)(n+3))/4 + (n+1)(n+2)(n+3)
- = (n(n+1)(n+2)(n+3))/4 + (4(n+1)(n+2)(n+3))/4
- = (n(n+1)(n+2)(n+3) + 4(n+1)(n+2)(n+3))/4
- = (n(n+1)(n+2)(n+3)(n+4))/4

9.

a) True
b) False
c) True
d) False

10.

a) Applying the implication law
- (~A ^ ~B) => ~B = ~(~A ^ ~B) V ~B

b) Applying DeMorgan's law
- ~(~A ^ ~B) V ~B = (A V B) V ~B

c) Applying idempotent law
- (A V B) V ~B = A V B

d) Applying idempotent law again
- A V B = True