Testing examples (using provided test samples):

Test1:

```
ern Sydney University\INFS3003 - AI\Ass 2\ass2.py"sity\INFS3003 - AI\Ass 2>
This is an extended propositional backward chaining system.
Enter facts to build knowledge base tree.
Type 'nil' to finish.

Enter facts: AvB=>E
Enter facts: A^B=>D
Enter facts: D^E=>F
Enter facts: B^E=>F
Enter facts: A
Enter facts: B
Enter facts: C
Enter facts: nil

See knowledge base nodes below:

Node E : [['A'], ['B']]
Node D : [['A', 'B']]
Node F : [['D', 'E'], ['B', 'E']]
Node A : [[]]
Node B : [[]]
Node C : [[]]

Test the system by entering a letter to be proven. Type 'quit' to exit

Prove: e
Yes, E can be achieved.
Prove: d
Yes, D can be achieved.
Prove: f
Yes, F can be achieved.
Prove: p
No, P cannot be achieved.
Prove: quit
PS C:\Users\louis\OneDrive - Western Sydney University\INFS3003 - AI\Ass 2>
```

Test2:

```
PS C:\Users\louis\OneDrive - Western Sydney University\INFS3003 - AI\Ass 2> pyt
ern Sydney University\INFS3003 - AI\Ass 2\ass2.py"
This is an extended propositional backward chaining system.
Enter facts to build knowledge base tree.
Type 'nil' to finish.

Enter facts: A^B^C^D=>F
Enter facts: BvEvG=>F
Enter facts: H^I=>A
Enter facts: J^K=>B
Enter facts: UvTvW=>J
Enter facts: I^J^T=>P
Enter facts: I^J^F=>Q
Enter facts: H
Enter facts: I
Enter facts: K
Enter facts: D
Enter facts: W
Enter facts: C
Enter facts: nil

See knowledge base nodes below:

Node F : [['A', 'B', 'C', 'D'], ['B'], ['E'], ['G']]
Node A : [['H', 'I']]
Node B : [['J', 'K']]
Node J : [['U'], ['T'], ['W']]
Node P : [['I', 'J', 'T']]
Node Q : [['I', 'J', 'F']]
Node H : [[]]
Node I : [[]]
Node K : [[]]
Node D : [[]]
Node W : [[]]
Node C : [[]]

Test the system by entering a letter to be proven. Type 'quit' to exit

Prove: f
Yes, F can be achieved.
Prove: a
Yes, A can be achieved.
Prove: b
Yes, B can be achieved.
Prove: j
Yes, J can be achieved.
Prove: p
No, P cannot be achieved.
Prove: q
Yes, Q can be achieved.
Prove: quit
PS C:\Users\louis\OneDrive - Western Sydney University\INFS3003 - AI\Ass 2> pyt
```

Test3:

```
PS C:\Users\louis\OneDrive - Western Sydney University\INFS3003 - AI\Ass 2> python -
ern Sydney University\INFS3003 - AI\Ass 2\ass2.py"
This is an extended propositional backward chaining system.
Enter facts to build knowledge base tree.
Type 'nil' to finish.

Enter facts: a^b=>c
Enter facts: cvd=>f
Enter facts: fvgvhvp=>x
Enter facts: x^y=>z
Enter facts: a
Enter facts: b
Enter facts: y
Enter facts: nil

See knowledge base nodes below:

Node C : [['A', 'B']]
Node F : [['C'], ['D']]
Node X : [['F'], ['G'], ['H'], ['P']]
Node Z : [['X', 'Y']]
Node A : [[]]
Node B : [[]]
Node Y : [[]]

Test the system by entering a letter to be proven. Type 'quit' to exit

Prove: z
Yes, Z can be achieved.
Prove: y
Yes, Y can be achieved.
Prove: f
Yes, F can be achieved.
Prove: x
Yes, X can be achieved.
Prove: z
Yes, Z can be achieved.
Prove: quit
PS C:\Users\louis\OneDrive - Western Sydney University\INFS3003 - AI\Ass 2> python -
```