verifyCNFs(listCNFs tests, 1);	
SENTENCE #1: CONJUNCTS: * (P)	======================================
SENTENCE #2: CONJUNCTS: * (-P)	* (P) ====================================
SENTENCE #3: CONJUNCTS: * (P)	CONJUNCTS: * (-P v Q) * (-Q v R) ====================================
SENTENCE #4: CONJUNCTS: * (-P) ====================================	SENTENCE #13: CONJUNCTS: * ((P v Q) v (-P v R)) * ((-Q v Q) v (-P v R)) * ((P v -R) v (-P v R)) * ((-Q v -R) v (-P v R))
* (P v -P) ====================================	SENTENCE #14: CONJUNCTS: * (P) * (Q) * (P v Q)
======================================	======================================
* (Q v -P) * (P v -Q) * (Q v -Q) ====================================	SENTENCE #16: CONJUNCTS: * (P v R)
SENTENCE #8: CONJUNCTS: * (-P v P) * (-Q v P)	* (Q v R) * (P v S) * (Q v S) ====================================
SENTENCE #9: CONJUNCTS: * (S v Q) * (T v Q) * (S v R) * (T v R) ====================================	SENTENCE #17: CONJUNCTS: * (P v (Q v -P)) * (-Q v (Q v -P))
	======================================
CONJUNCTS: * (-S) * (-T)	

```
SENTENCE #19:
CONJUNCTS:
* ((-P \vee Q) \vee P)
* ((Q v - P) v P)
* ((-P \vee Q) \vee -Q)
* ((Q v - P) v - Q)
* ((-P v Q) v -Q)
* ((Q v - P) v - Q)
* ((-P \vee Q) \vee P)
* ((Q v - P) v P)
SENTENCE #20:
CONJUNCTS:
* (P v (P v Q))
* (O v (P v O))
* ((-P v - Q) v - P)
* ((-P v - Q) v - Q)
______
SENTENCE #21:
CONJUNCTS:
* (P)
* (Q)
* (P v Q)
SENTENCE #22:
CONJUNCTS:
* ((P \vee Q) \vee (-P \vee R))
* ((-Q \vee Q) \vee (-P \vee R))
* ((P v - R) v (-P v R))
* ((-Q v - R) v (-P v R))
_____
SENTENCE #23:
CONJUNCTS:
* ((-P \vee Q) \vee (-Q \vee R))
_____
SENTENCE #24:
CONJUNCTS:
* ((P \vee Q) \vee (-P \vee R))
* ((-Q \vee Q) \vee (-P \vee R))
* ((P v - R) v (-P v R))
```

* ((-Q v - R) v (-P v R))

SENTENCE #25:

CONJUNCTS:

- * ((P v P) v (S v Q))
- * ((Q v P) v (S v Q))
- * ((P v R) v (S v Q))
- * ((Q v R) v (S v Q))
- * ((P v P) v (T v Q))
- * ((Q v P) v (T v Q))
- * ((P v R) v (T v Q))
- * ((Q v -R) v (T v -Q))
- * ((P v P) v (S v P))
- * ((Q v P) v (S v P))
- * ((P v R) v (S v P))
- * ((Q v -R) v (S v -P))
- * ((P v P) v (T v P))
- * ((Q v -P) v (T v -P))
- * ((P v R) v (T v P))
- * ((Q v R) v (T v P))

val it = () : unit

-