

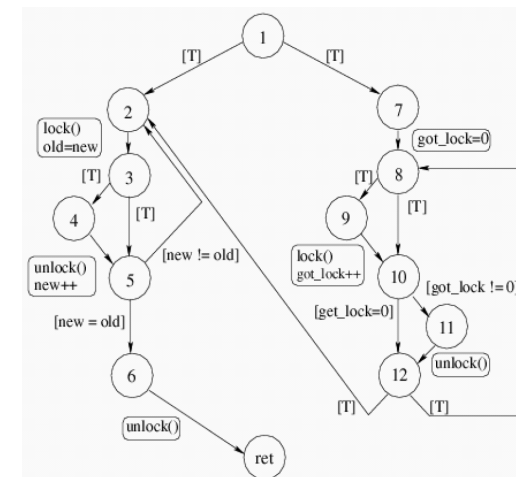
Introduction à la **vérification**

La **fiabilité logicielle** est un enjeu **majeur**



Approche: **modéliser** les parties critiques des **programmes**

```
Example() {
1:  if (*){
7:    do {
        got_lock = 0;
8:      if (*){
9:        lock();
        got_lock++;
        }
10:     if (got_lock){
11:      unlock();
        }
12:    } while (*)
2:  } do {
        lock();
        old = new;
3:    if (*){
4:      unlock();
        new++;
        }
5:  } while (new != old);
6:  unlock();
   return;
}
```



Introduction à la **vérification**

Prérequis : automates finis, algorithmique de graphes, bases de C

Contenu :

- modéliser les programmes comme des automates
- exprimer les propriétés des programmes (LTL, CTL)
- utilisation d'un outil (SPIN ou UppAal)

Organisation : 9H CM, 10H TD, 10H TP

Evaluation :

- Contrôle continu : mini-projet
- Examen terminal : 2h
- $NF = 0,2*CC + 0,8*ET$