

Curs 9 - Sisteme de operare

Settings → system → 4 CPU's

FPGA

libpthread.a
libm.a

-S la compilare

-O1 -O2 -O3

- doar unul are acces (excludere mutuala) → mutex
- doar cel care modifica are acces exclusiv → rwlock
- un număr limitat să aibă acces → semafor
- ~~sa poartim / continut~~ sa poartim / continut în tot ~~otot~~ → bariera
- ~~stai până ești anunțat~~ → variabilă condițională
- stai până ești anunțat dar poartă în anumite condiții → variabilă cond.

semget — sem-init
semop — sem-wait
semctl — sem-post
 — sem-destroy

semafor: valoare ... 1 → semafor binar = lock
wait (rezerva) —————→
post (eliberează) —————→ unlock

bariera: valoare 3 și poartă dublu toate dau wait

variabilă condițională → signal pt. unul care este apt.
 → broadcast se trezesc toate care așteaptă.

Worker

```
if (login)
    lock();
    n++;
    if (n > 1000) {
        signal();
    }
    unlock();
}
```

Killer
while(1) {
 wait();
 //curator
}

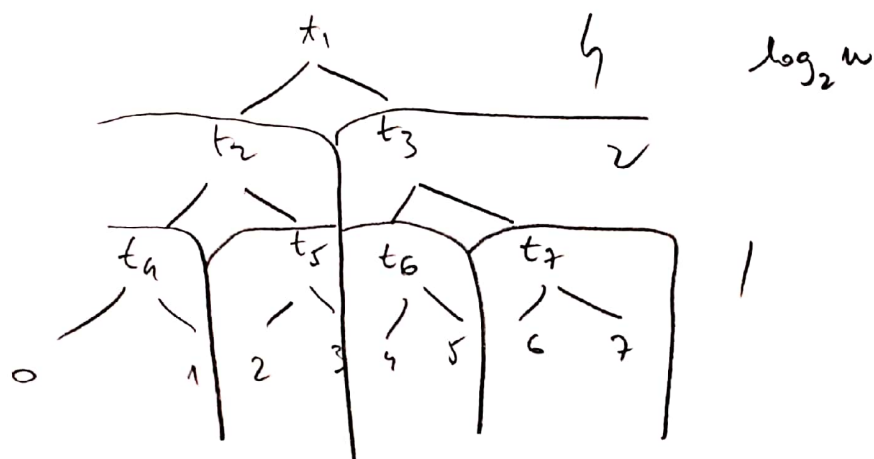
```

var count c;
int x;
mutex m;

```

<p>W</p> <pre> lock(&m); x++; if(x > 100) { signal(&c); } unlock(&m); </pre>	<p>K</p> <pre> while(1) { lock(&m); while(x <= 1000) { wait(&c, &m); } //curator unlock(&m); } </pre>
---	--

→ de ce wait deblochează mutex?



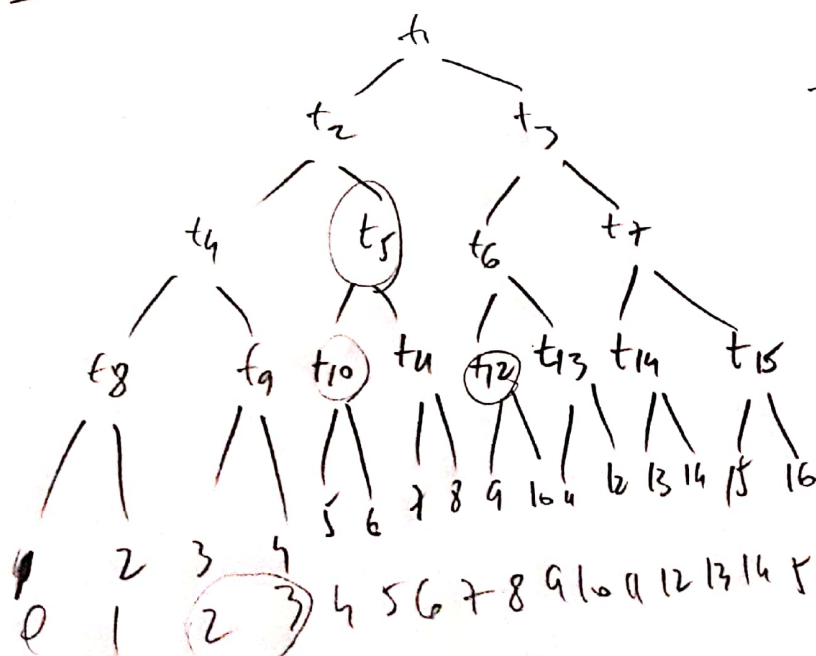
$$t_2 - 0 \leq 2$$

$$t_3 - 4 \leq 6$$

$$t_1 - 0 \leq 4$$

0	0
1	2
2	4
3	6

0	0
1	4



$$t_5 - 5$$

$$n = 17$$

$$init = 174$$