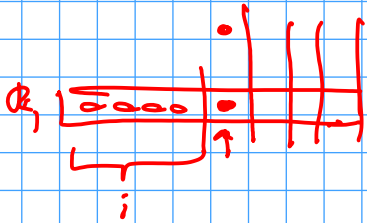
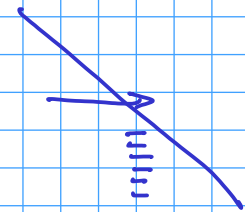


$R_i$

$R_j$



$$R_j = a R_i + b R_j$$

$$R_{j'} = \dots$$

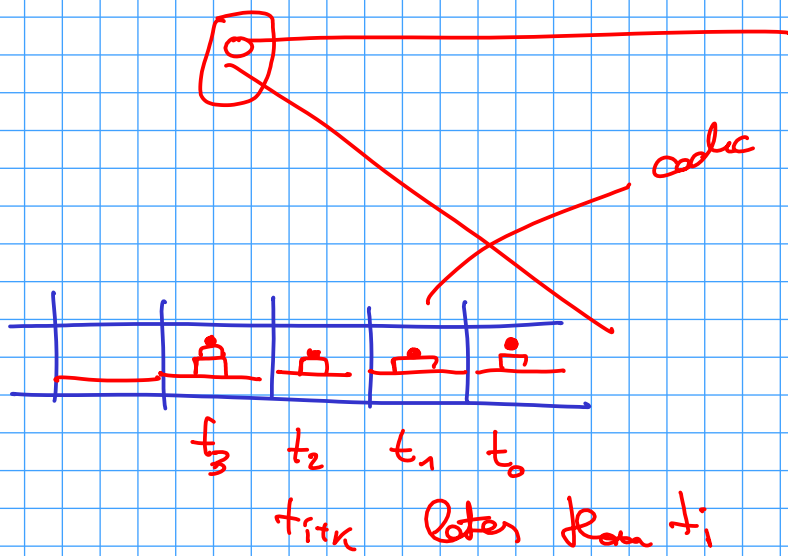
$W(R_j)$

$R(R_i, R_j)$

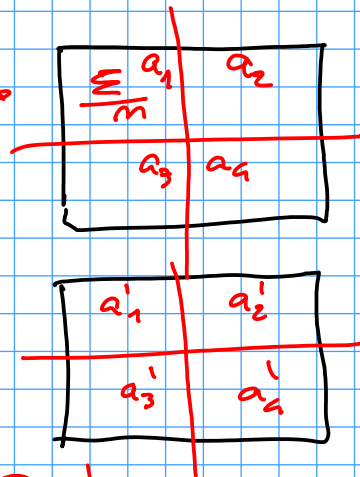
$W(R_{j'})$

$R(R_{i'}, R_{j'})$

# Video intrusion detection

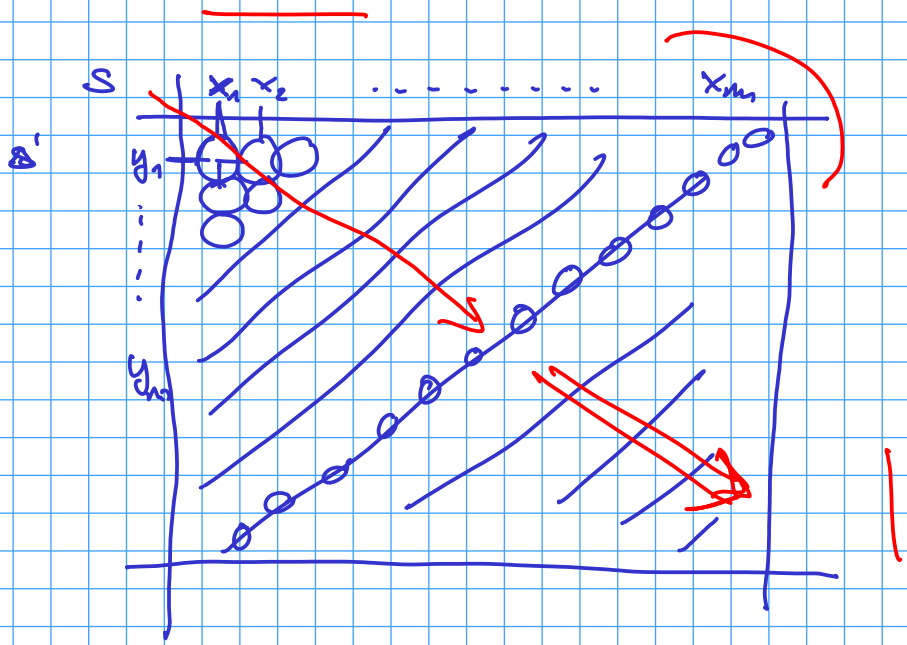
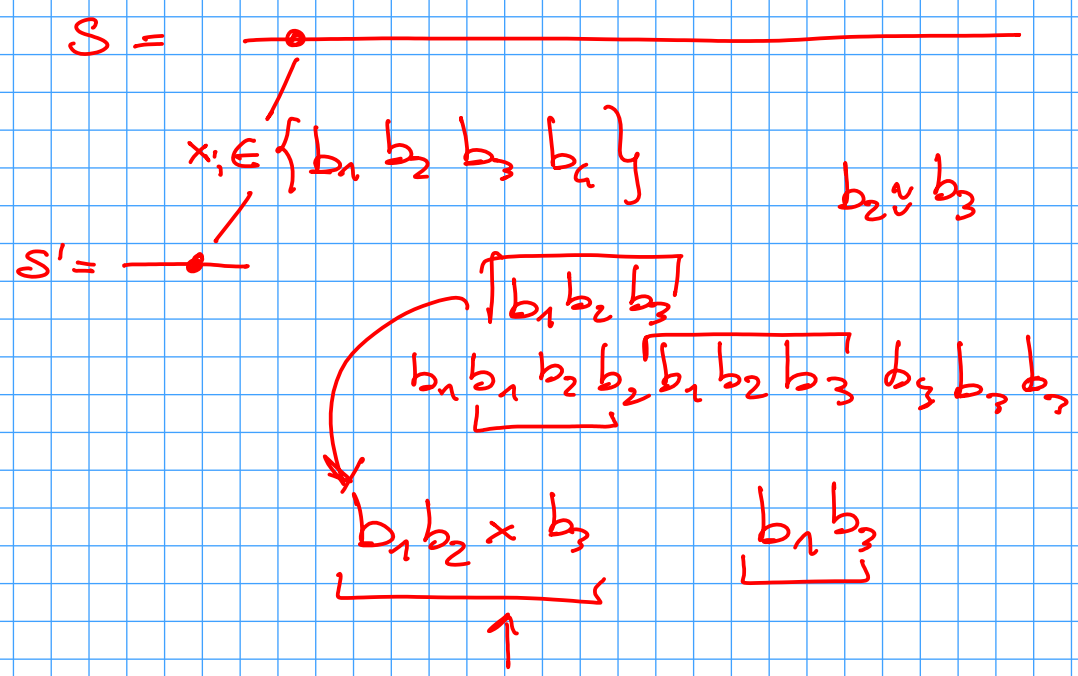


codebook



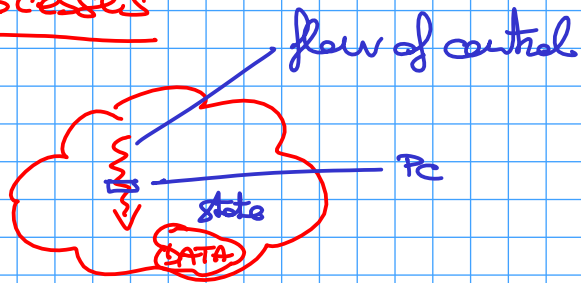
$$\forall_i \left\{ |a_i - a'_i| < \epsilon \right\}$$

genome

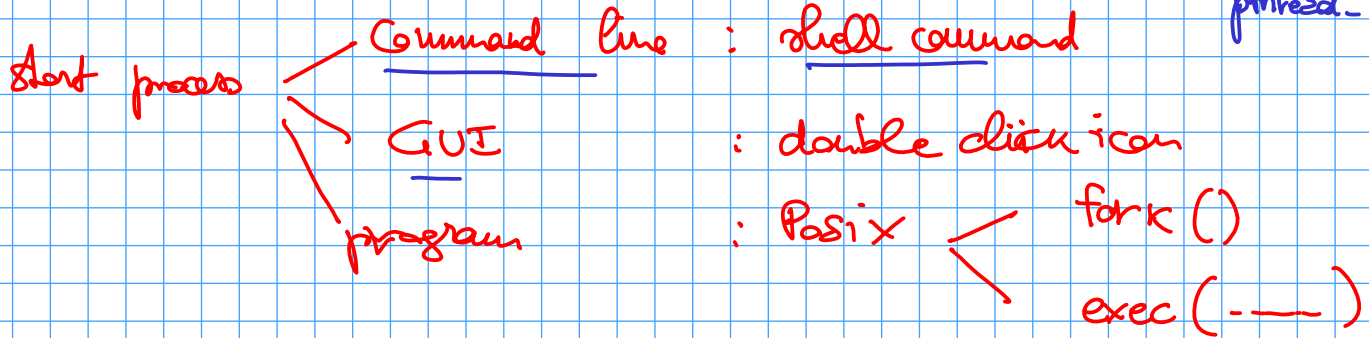
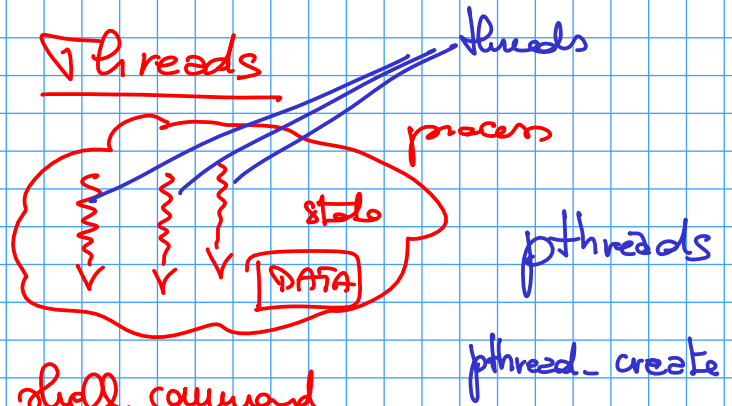


reunis concurent echites  
 communication mechanisms  
 synchronization mechanisms

# Processes

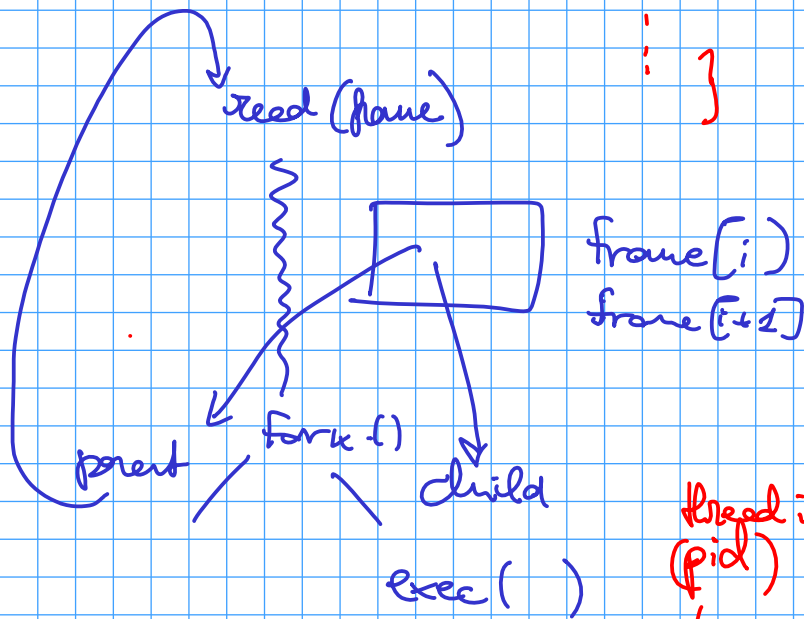


# Threads



+ wait  
syscall

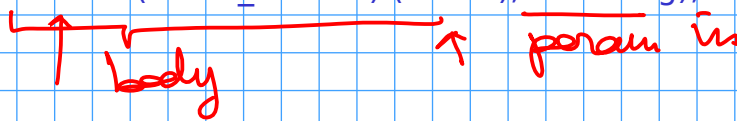
```
if ( (pid = fork()) == 0 ) {
    execv(.....);
} else {
    ...
}
```



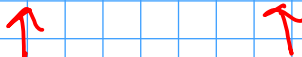
thread id  
(pid)

attributes

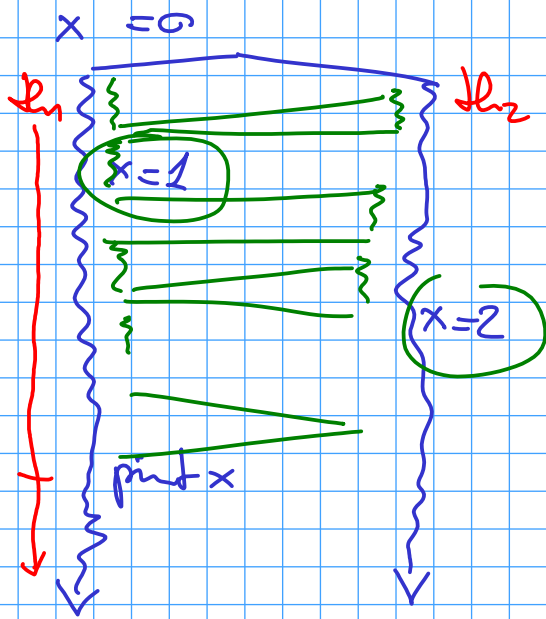
```
int pthread_create(pthread_t *thread, const pthread_attr_t *attr,
void *(*start_routine)(void *), void *arg);
```



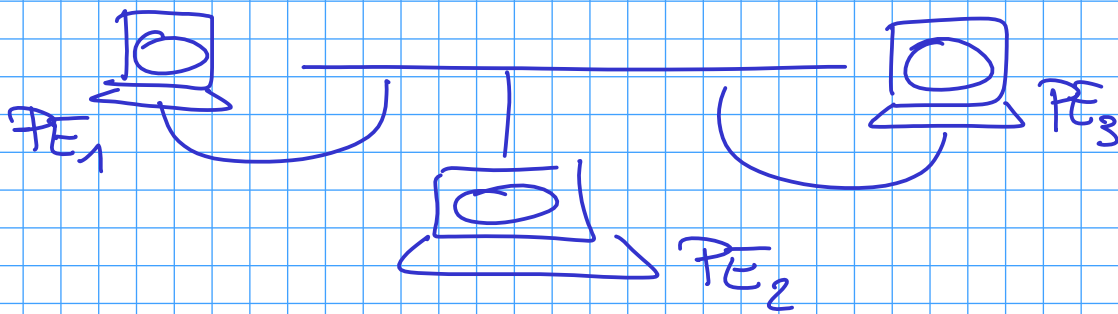
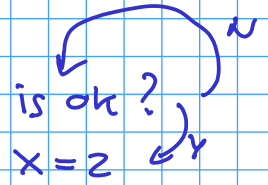
```
int pthread_join(pthread_t thread, void **retval);
```



lib  
calls



$x=1$   
 tell  $h_2$  ok



$S_{xx}$  }  $ssh$   
 scp

$> \text{cmd}_1 p_1 \dots p_n$   
 ssh  $\uparrow$   $\text{cmd}_1 p_1 \dots p_n$   
 hostname

