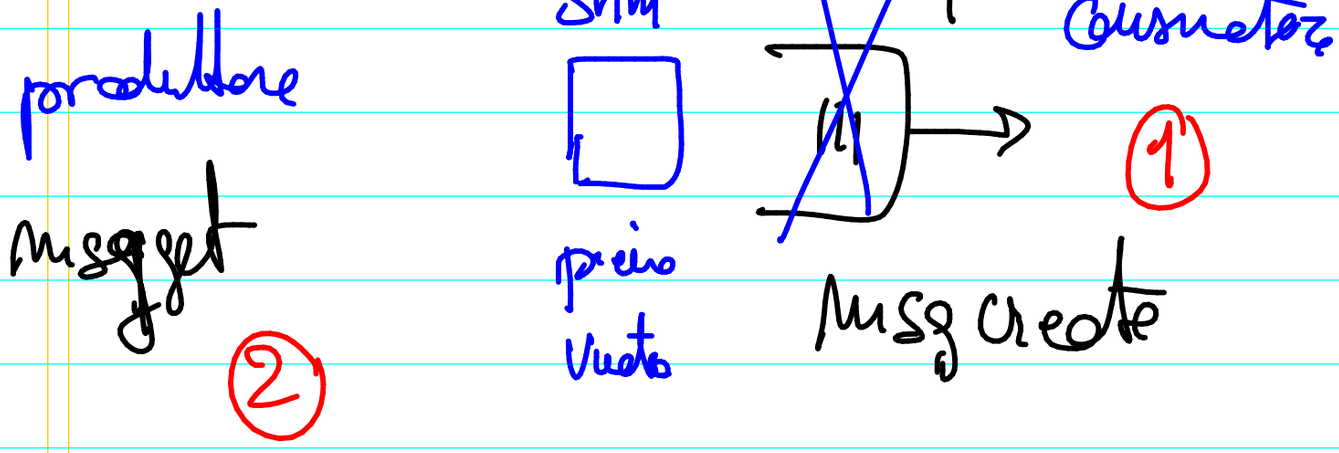
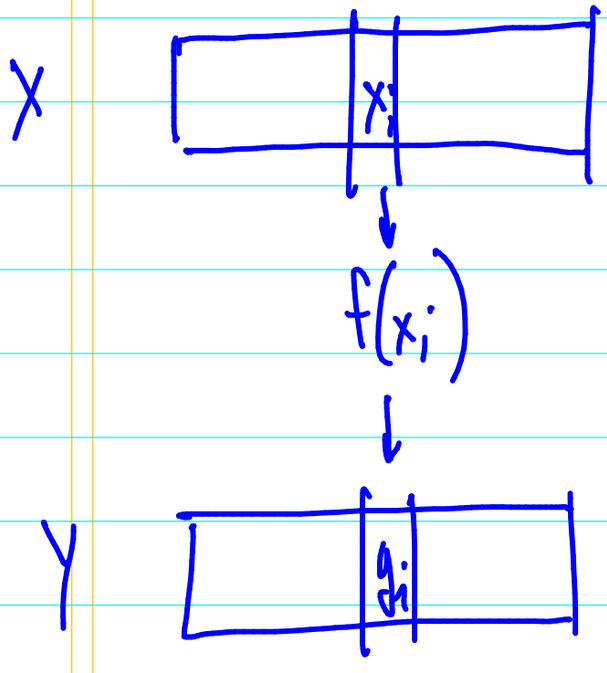
SYS

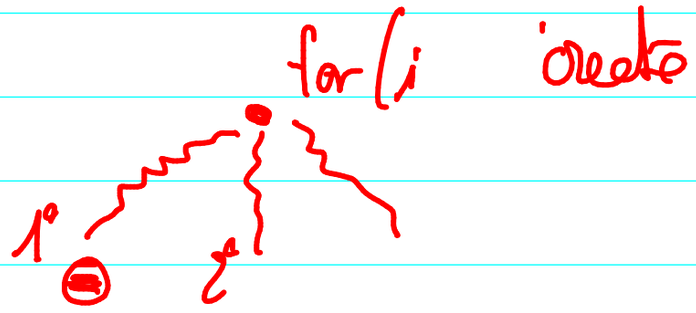
msg
shmem
sem





embarrassingly parallel

owner computes rule



OpenMP #pragma

```
#pragma omp parallel for (chunksize = 10)
for (int i = 0; i < NTASKS; i++) {
    int res = Worker(data[i]);
    ...
}
```

chunksize

core = 4

i=0 i=1	.....	i=NTASKS-1
chunk1	chunk2	chunk3	chunk4

↑ ↑

-fopenmp

Main

```
# pragma omp parallel for
for( ) {
    :
    pthread_create(....)
}
```

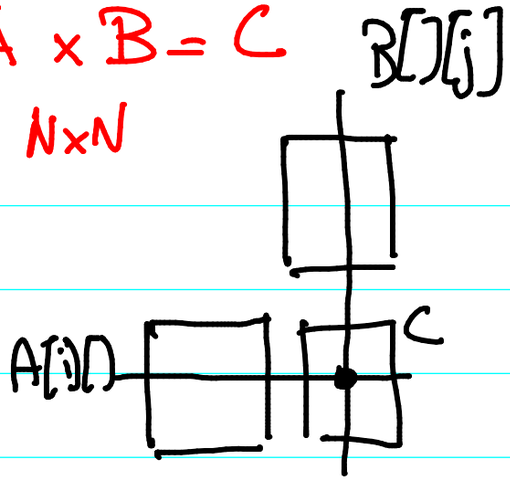
```
main( ... ) {

}
# pragma omp task
{ blocco di codice }
```

```
# prog
for(i=0; i<N; i++)
  for(j=0; j<N; j++) {
    C[i][j] = 0;
    for(k=0; k<N; k++)
#     C[i][j] += A[i][k] * B[k][j];
  }
```

$A \times B = C$

$N \times N$



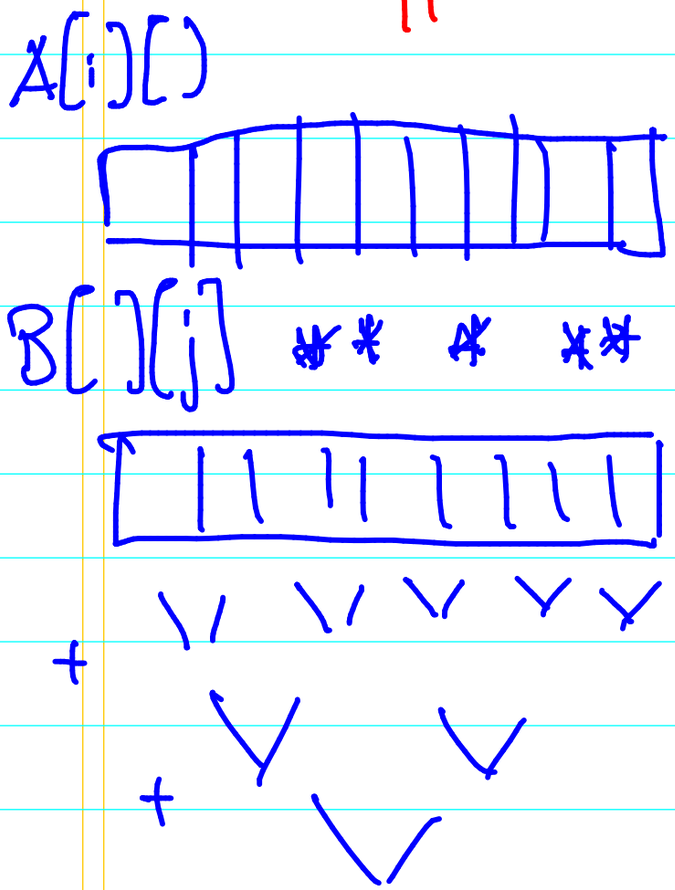
CONCETTUALMENTE

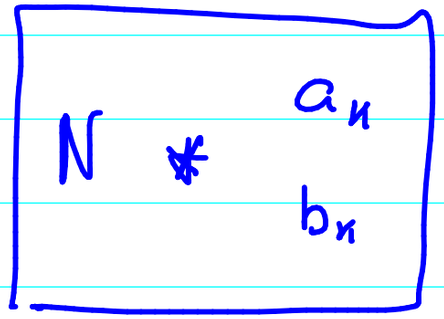
MAX PAR.

E il calcolo

C[i][j] += ...

IP

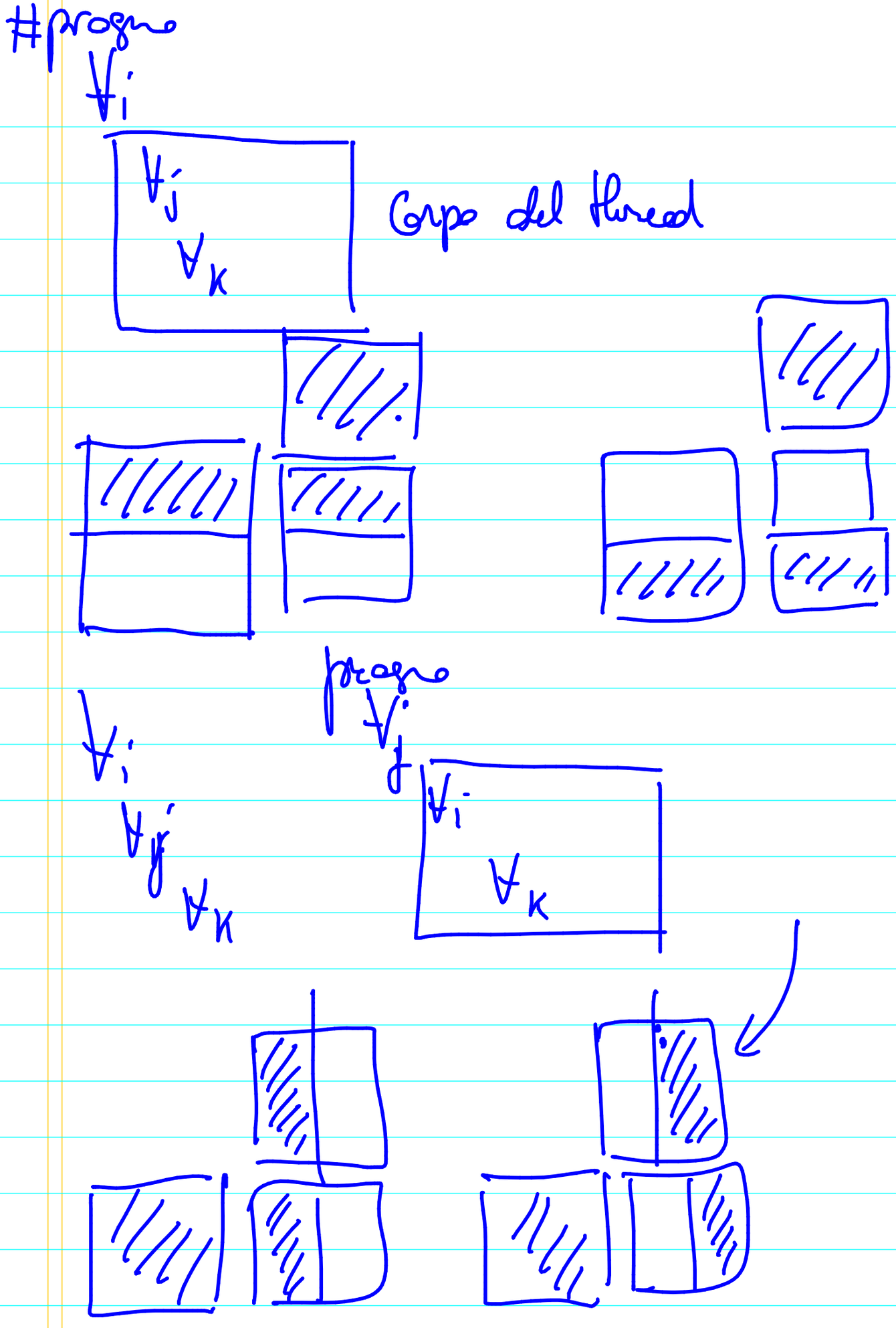




O(1)

+ O(N)

map reduce


#prag
Corpo del thread
prag

$\oplus$ assoc. & comm

MAP REDUCE

↓

↳ "Sommare" tutti gli elementi di una collezione con $\oplus$

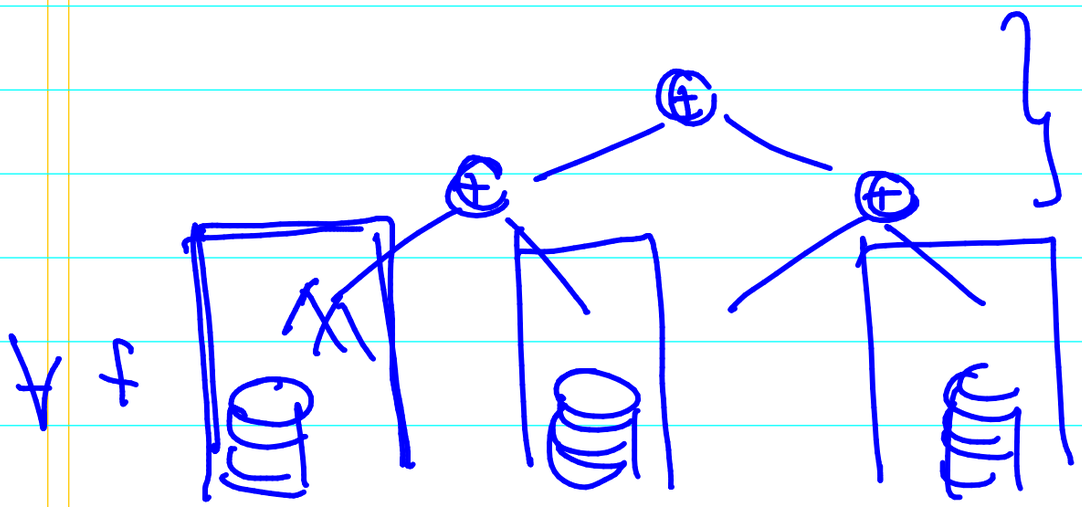
Calcolare f su tutti gli elementi di una collezione

$$\forall i \quad x_i \rightarrow f(x_i)$$

GOOGLE MAP REDUCE

$$\forall i \quad f(x_i) \rightarrow \langle k, v \rangle$$

$$\forall k_j \qquad \Sigma_{\oplus} \{ v \mid \langle k_j, v \rangle \}$$



1977 John Backus (FP)

$\langle a, b, c \rangle$ $\quad [f, g] : x$

zip $\quad \langle\langle a,b\rangle \langle c,d\rangle\rangle \quad \langle fx, gx\rangle$

$\langle\langle a,c\rangle, \langle b,d\rangle ..\rangle$ map $\quad \alpha f : \langle x_1 .. x_n\rangle = \langle f x_1 \cdots f x_n\rangle$

rotl
rotr

reduce $\quad /+ : \langle x_1 \cdots x_n\rangle = x_1 + x_2 \cdots + x_n$

$\bar{1}$ $\quad$ distl $\quad \langle a, \langle x_1 \cdots x_n\rangle\rangle = \langle\langle a, x_1\rangle .. \langle a, x_n\rangle\rangle$

$$\alpha\alpha(IP) \circ \alpha\, distr \circ \alpha\, distl \circ [\bar{1}, trans \circ \bar{2}] : \langle A, B\rangle$$

$$IP = \langle right; dft \rangle \langle left; B\rangle$$

$$IP = /+ \circ \alpha * \cdot zip$$