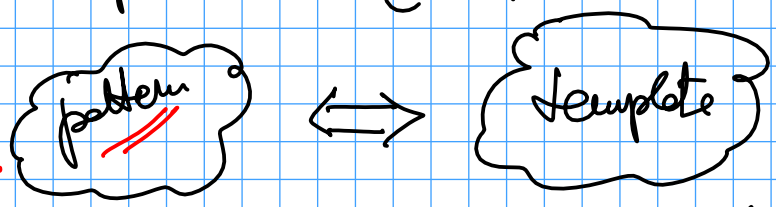
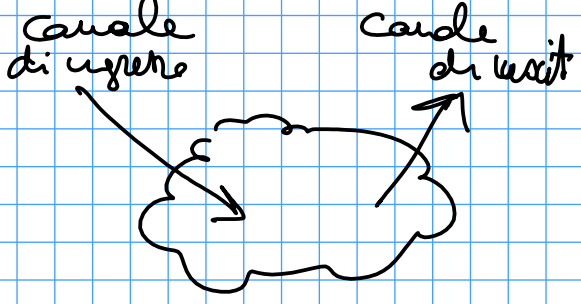
① Template × ordinazione × skeleton/pattern

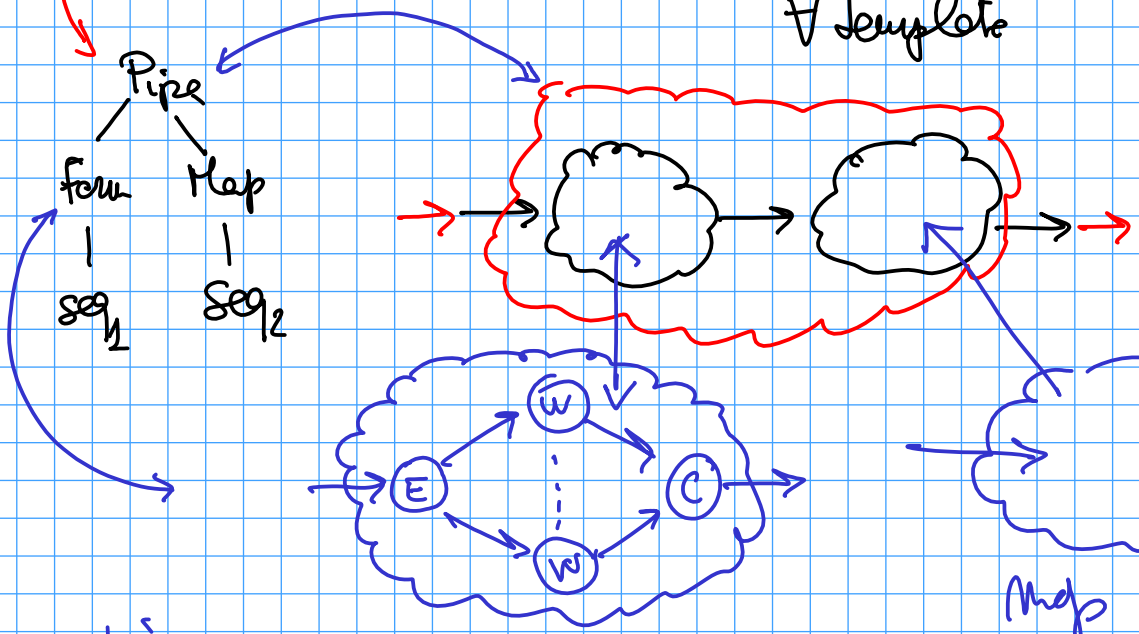
② Componibili (template)



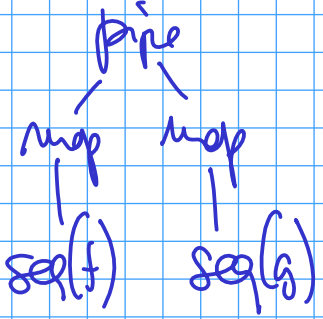


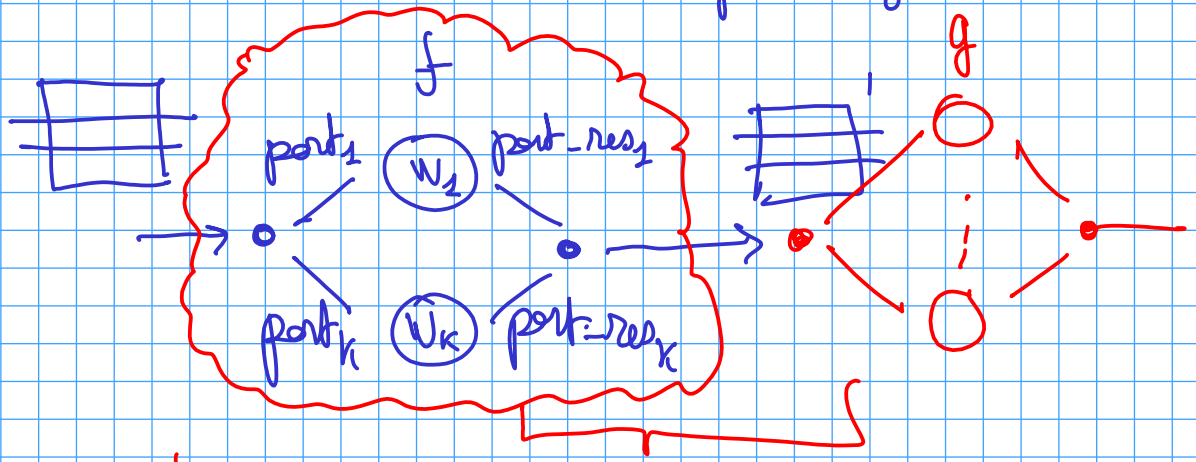
$port_j$ buf_j

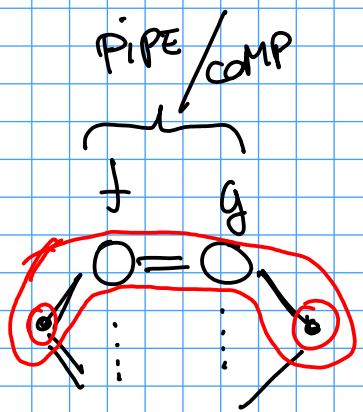
$\forall$ template

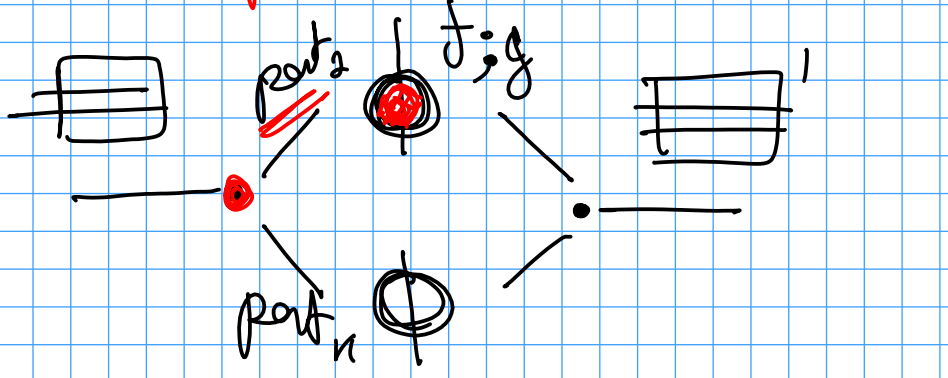


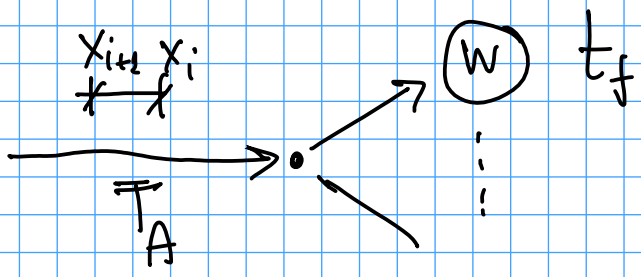
Map buf_j

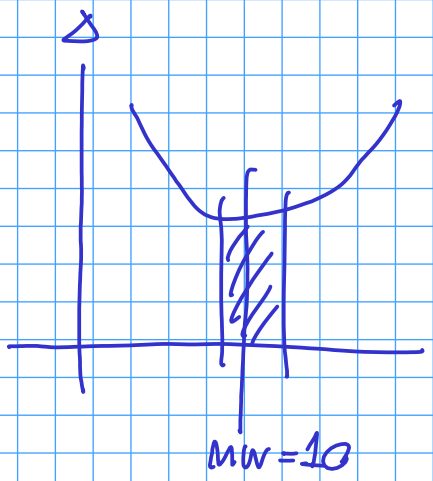




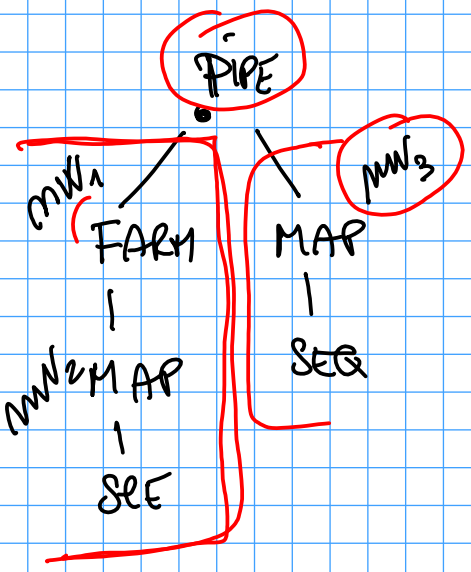








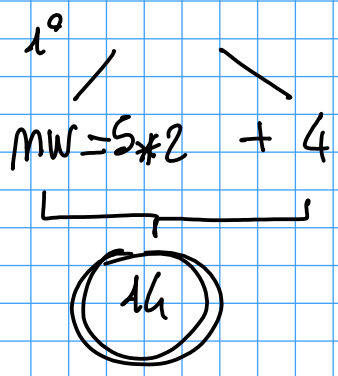
$nw = 10$



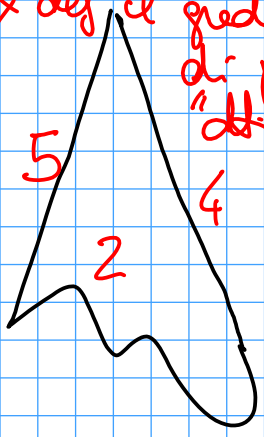
$nw_1 = 5$

$nw_2 = 2$

$nw_3 = 4$



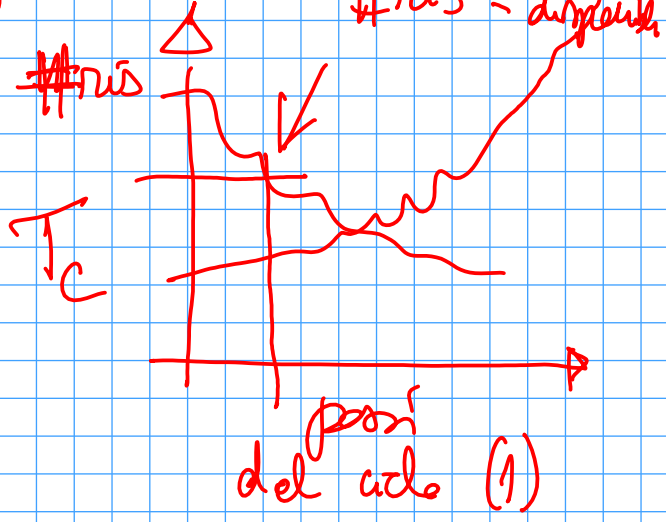
modelli analitici x def il grado di par. "ottimo"



$\sum nw >$ #risorse disponibili

CICLO (1)

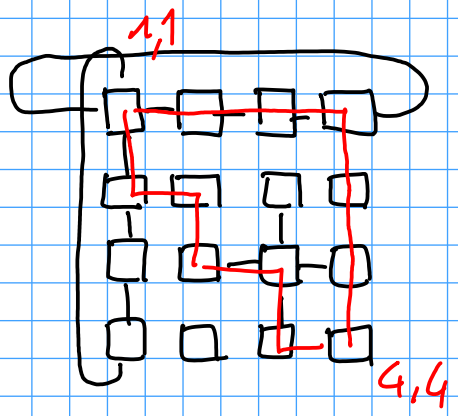
riduce il minimo numero di risorse mantenendo l'equilibrio fra le varie parti secondo i nuovi modelli analitici

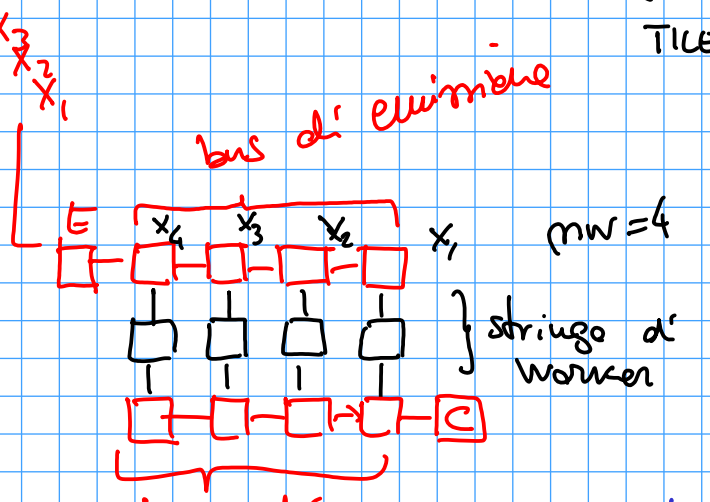


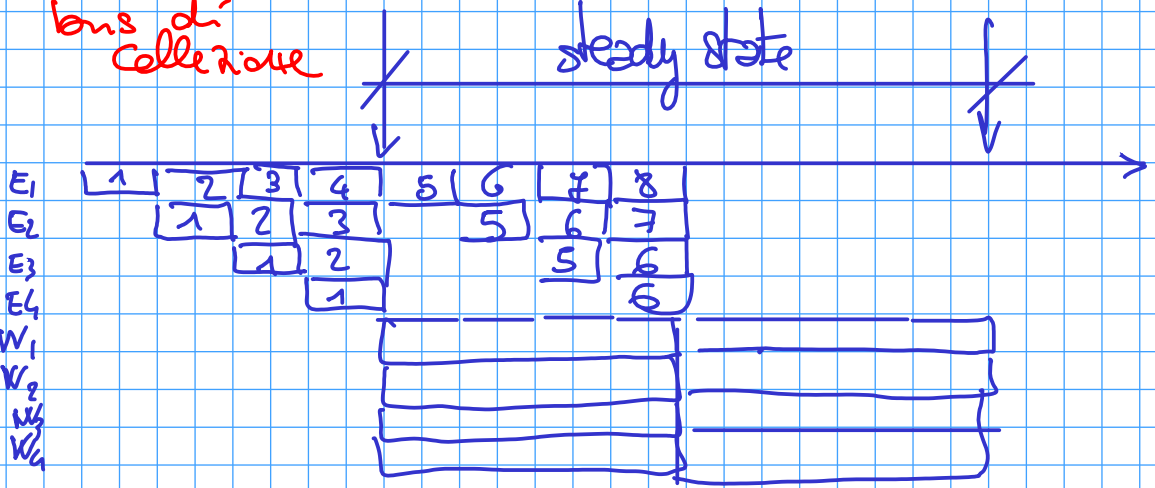
post del ciclo (1)

TEMPLATE × FARM (architetture a MESH)

TILERA



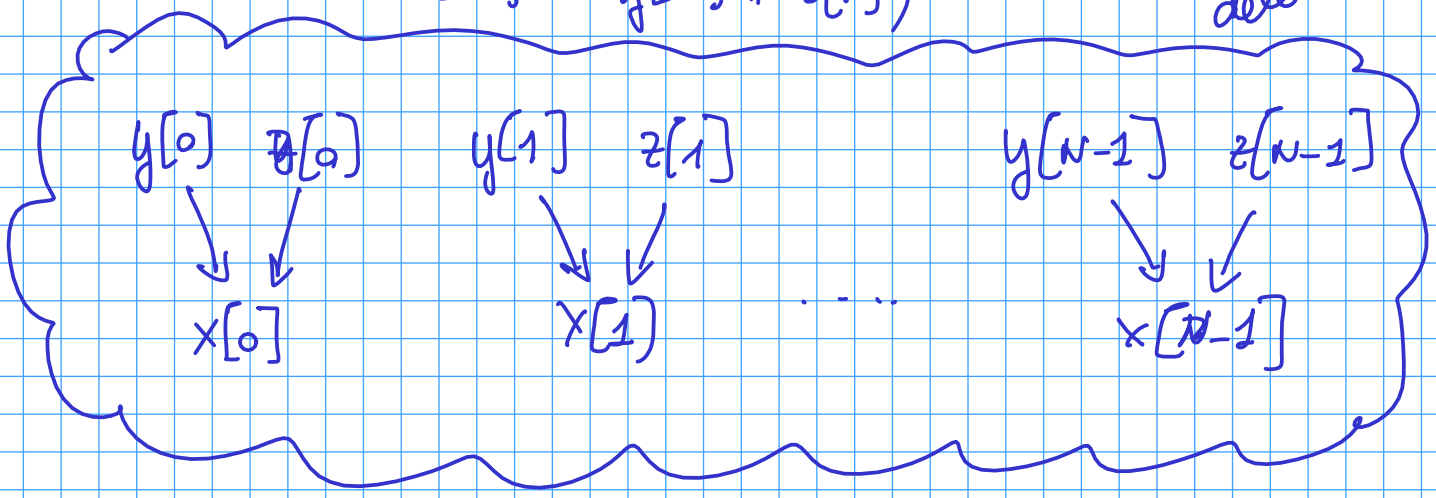


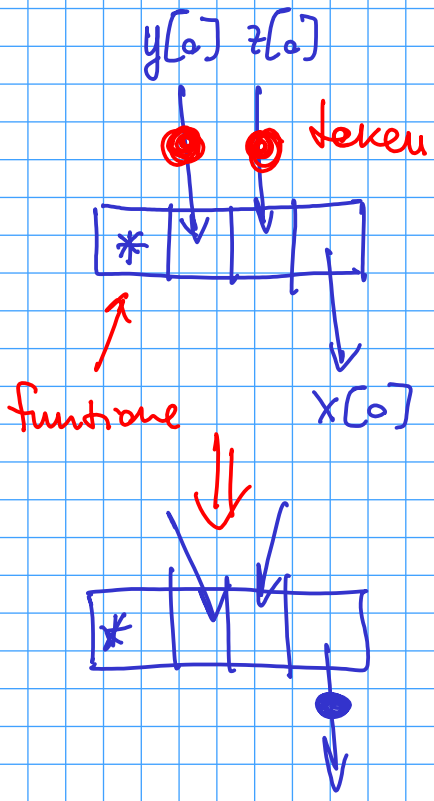


(MACRO) DATA FLOW

```
for(i=0; i<N; i++)
    x[i] = y[i] * z[i];
```

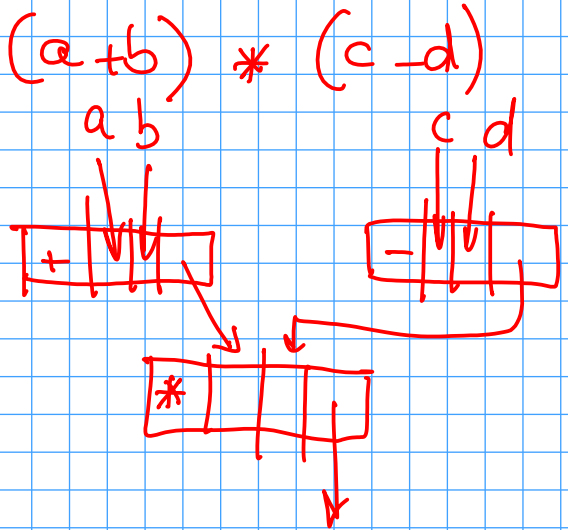
GRAFO DATA FLOW della computazione





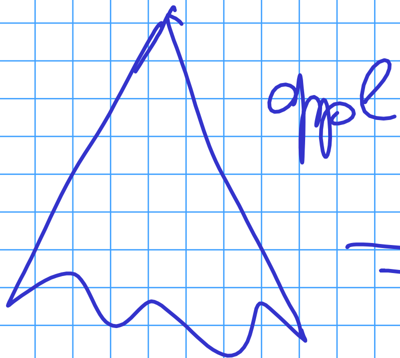
ISTRUZIONE DATA FLOW

è fireable sse sono presenti tutti i token di input



pattern → compilazione dei pattern in DATA FLOW (MACRO)

⇓



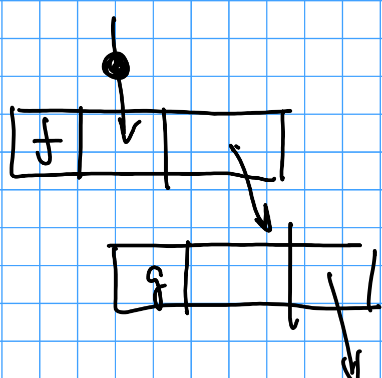
⇒ grafo MDF

⇓

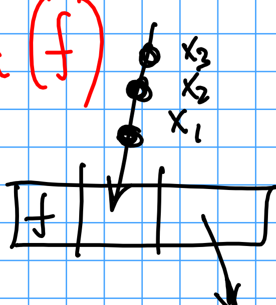
interprete distribuito/parallelo MDF

CICLO: trova le istr. fireable e le schedula x l'esecuzione
∀ risultato dirige i token alle istruzioni target

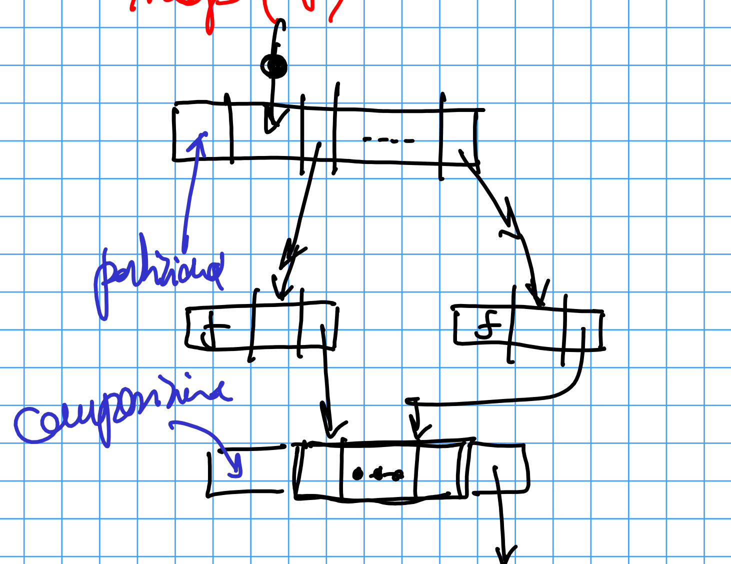
pipeline (f, g)

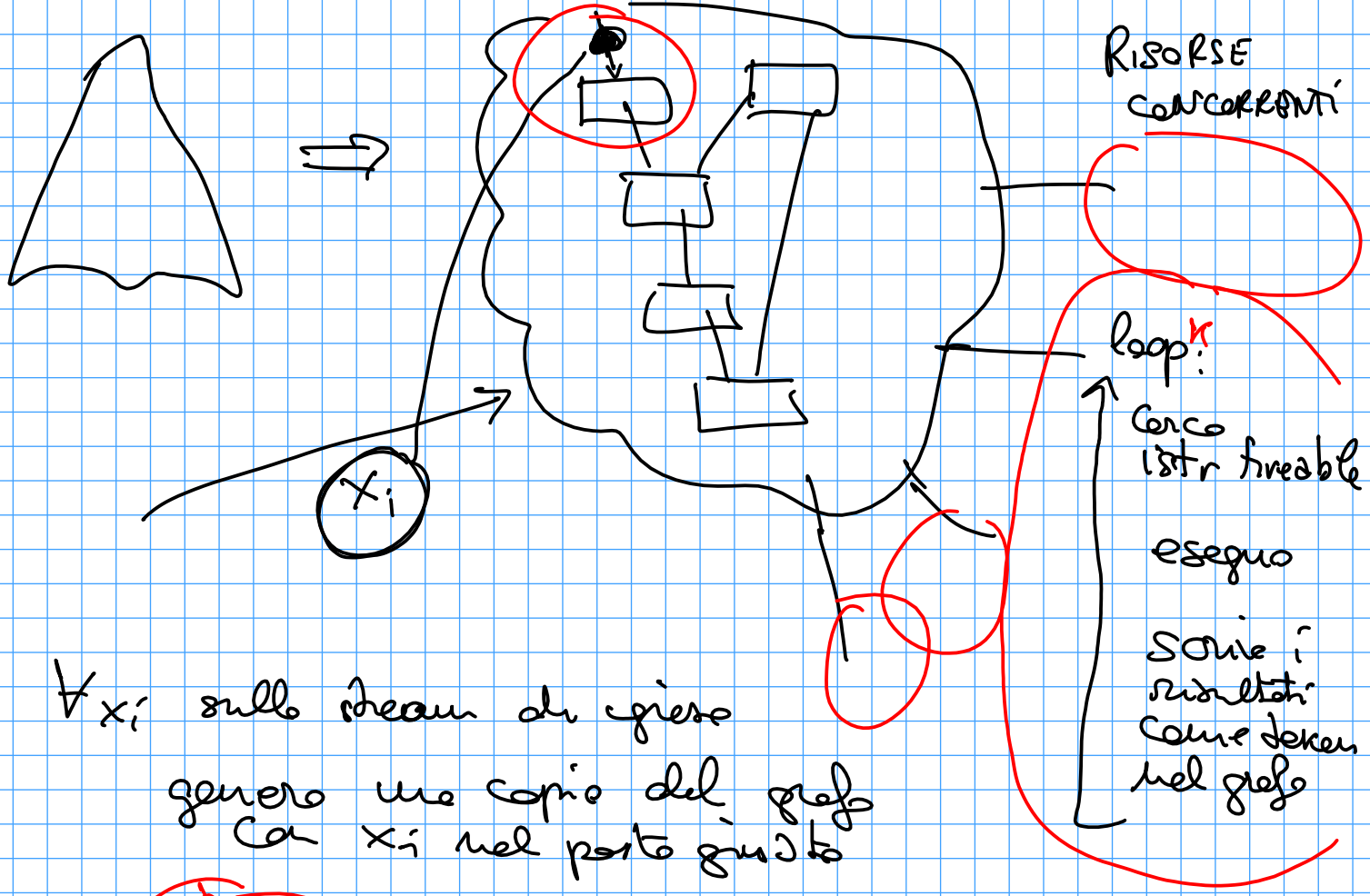


farm (f)



map (f)





$\forall x_i$ sulla stream di ingresso

genero una copia del grafo con x_i nel posto giusto