

Code Splitwise

Somnio

① Algos to settle up

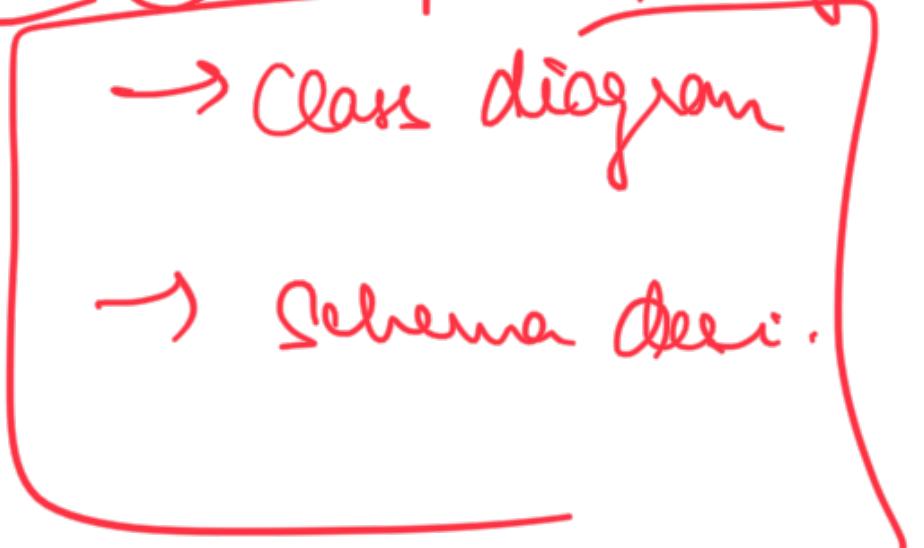
⇒ ② How to handle settle up?

(will be
starting at
9:10)

ISMnit

③ Code Walkthrough

30mn ④ Implementing the Command



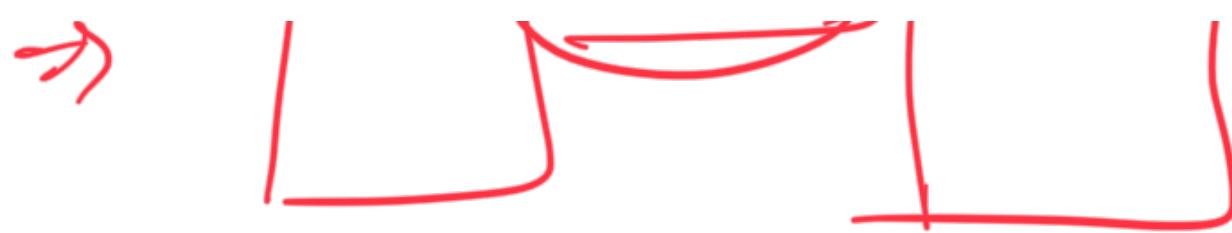
Father

Expense . getParticipants

① expense
② $\{ \leftarrow < \text{participants} \rightarrow \}$

Expense
 $\{ \leftarrow \text{list} < \text{participants} \rightarrow \}$

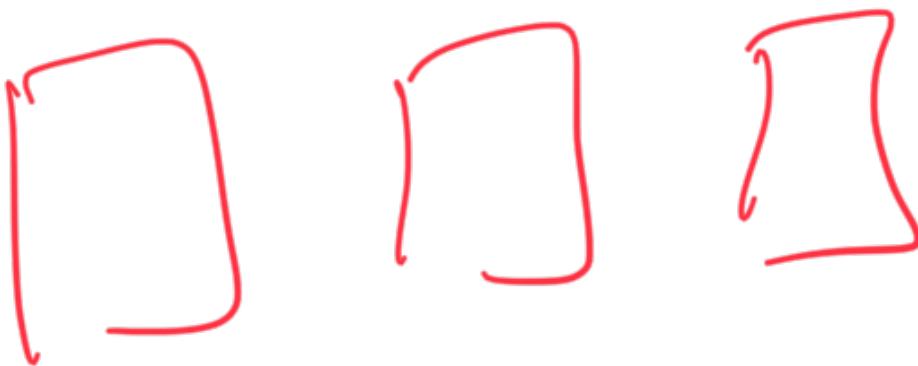
1 min:m)



PulseCell

Semaphores

Distributed Cache



Support @ Loker-on



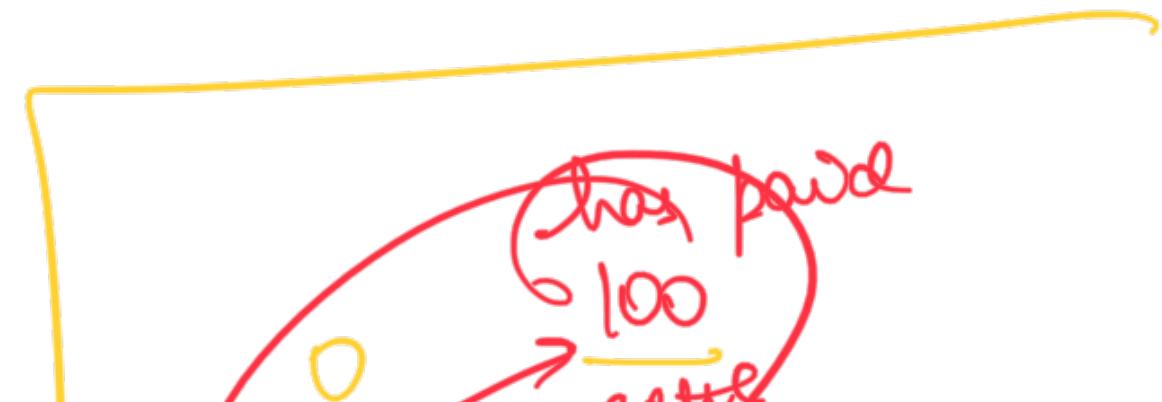
movie recommender system

Relations

1

Group

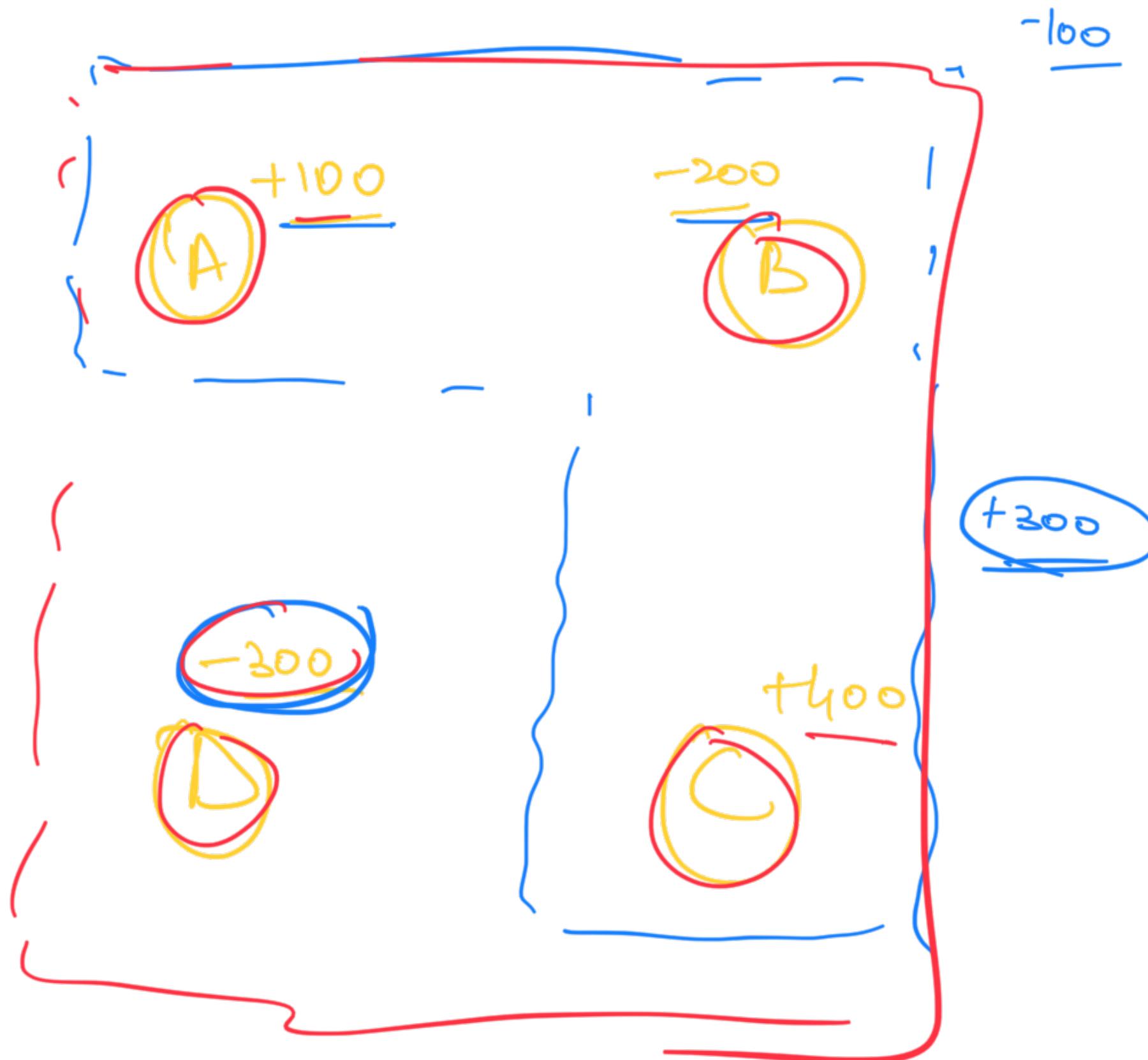
Settle up

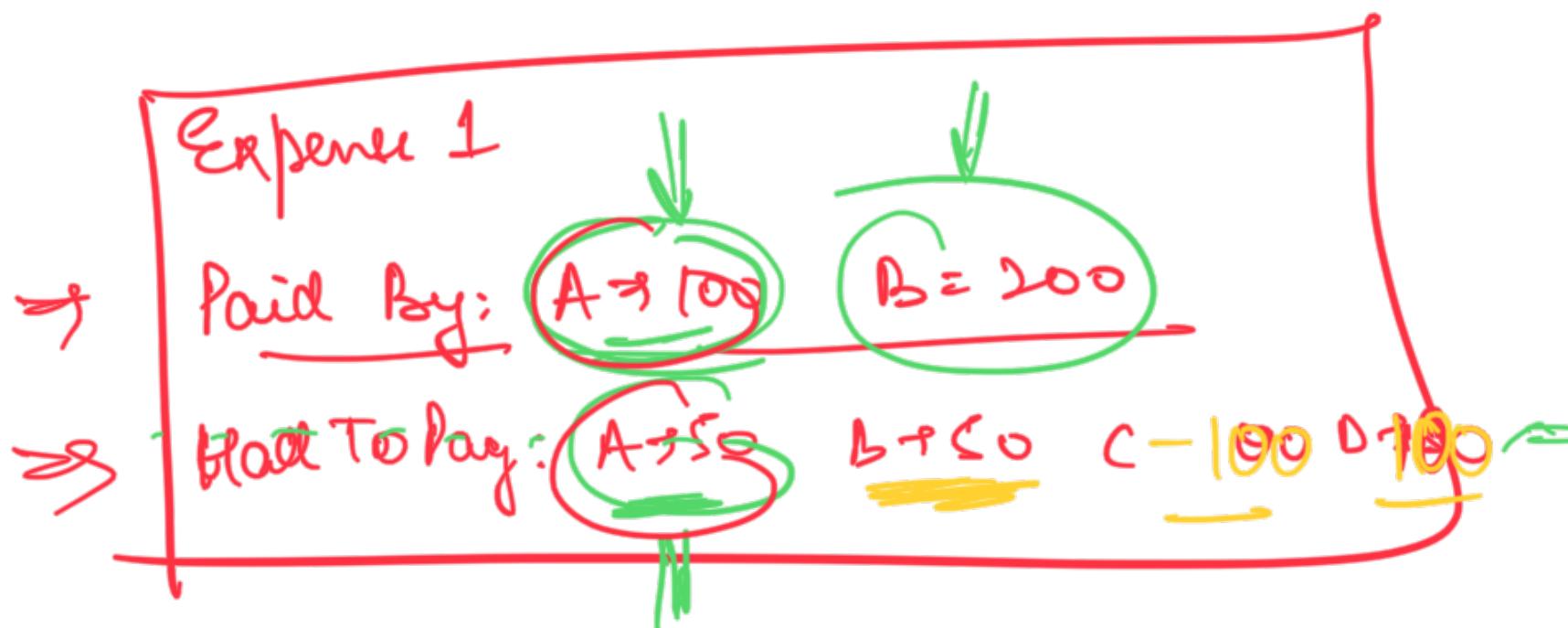




II

$A \rightarrow 200$	\Rightarrow pair
$B \rightarrow 100$	$A \rightarrow 100 \Rightarrow$ had to pay





Paid By - Had To Pay

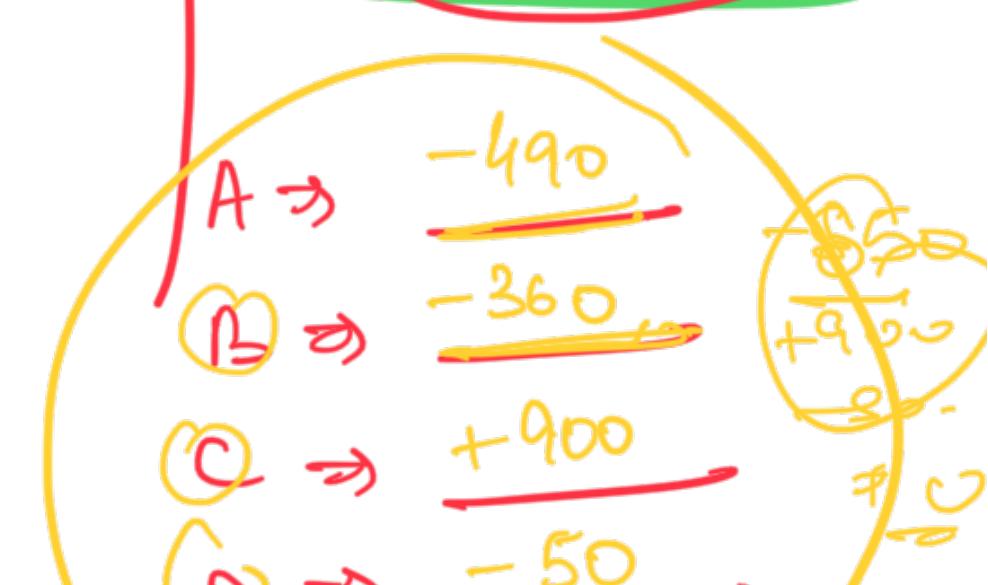
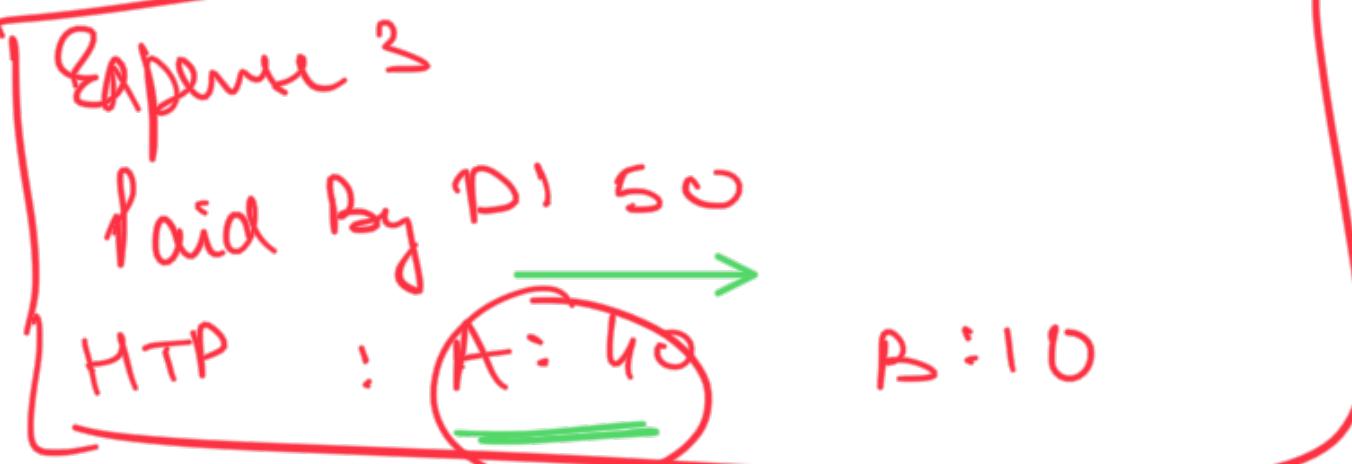
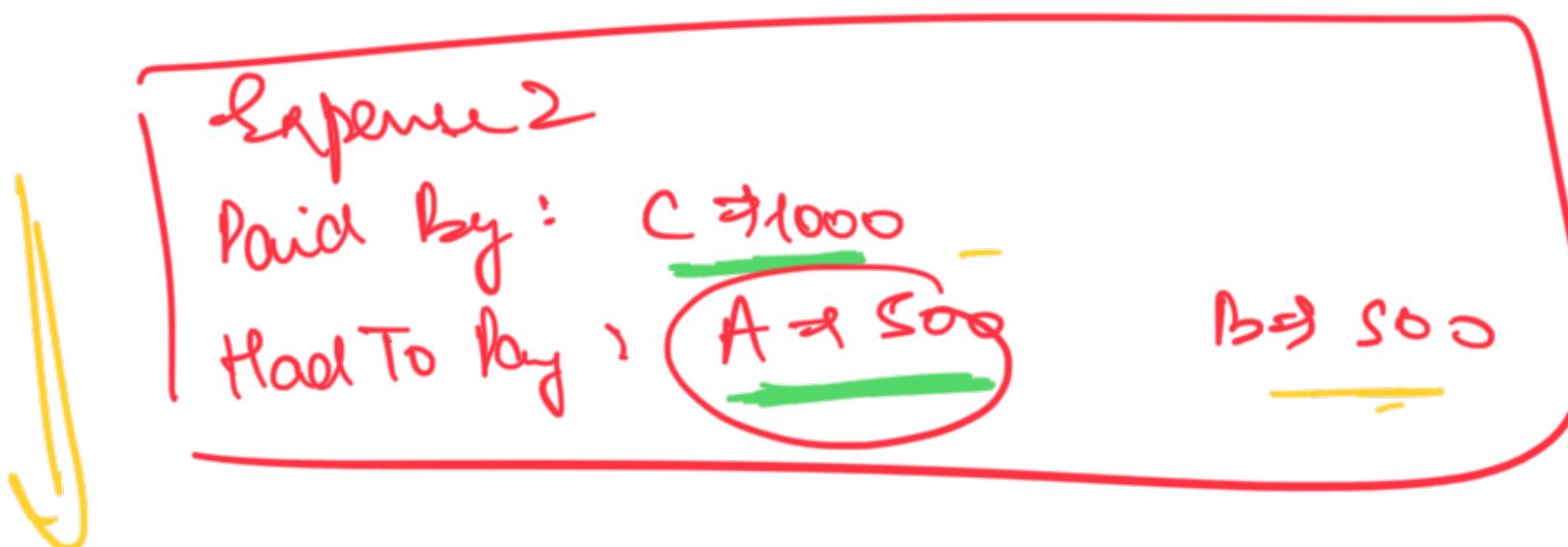
(A)

$$\begin{array}{r}
 +100 \\
 -50 \\
 \hline
 +0
 \end{array}$$

$$\begin{array}{r}
 -500 \\
 +0 \\
 \hline
 \end{array}$$

$$\begin{array}{r}
 -40 \\
 \hline
 \end{array}$$

(-490)



$$B: \begin{array}{r} +200 \\ -50 \\ \hline +0 \\ -500 \\ \hline +0 \\ -10 \\ \hline \end{array}$$

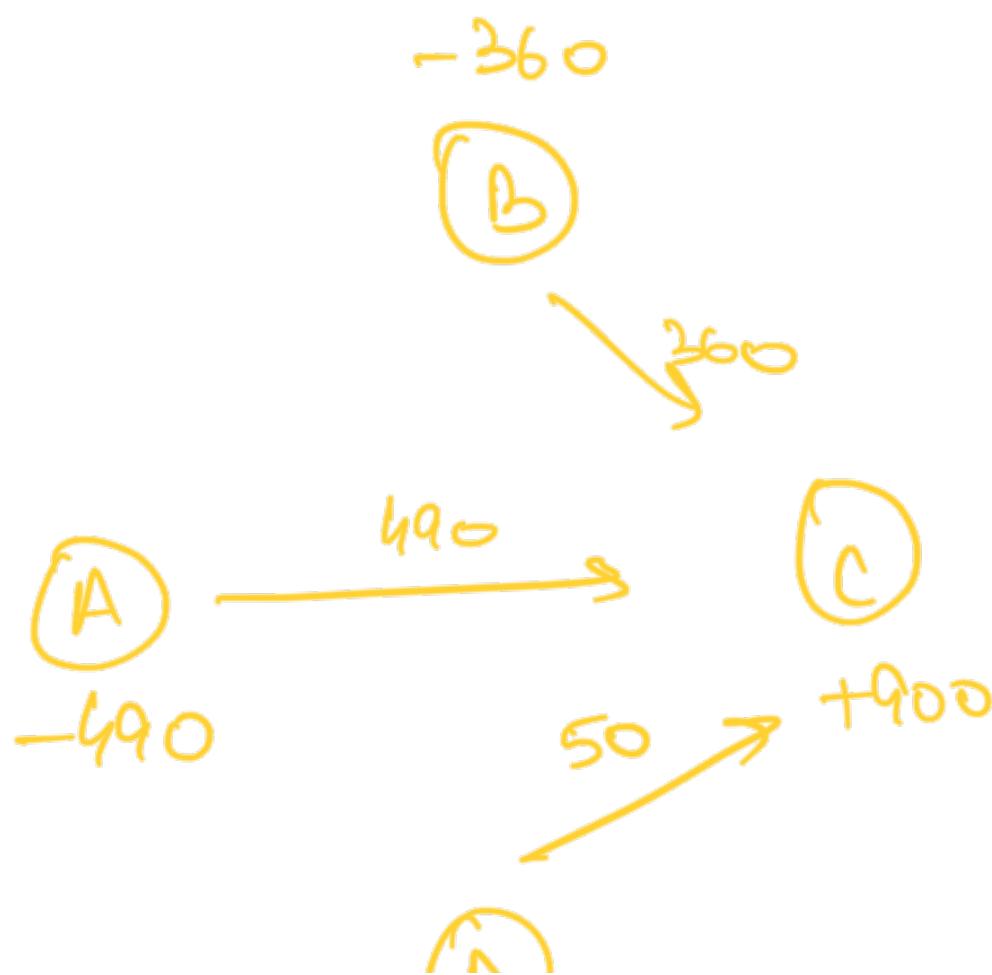
~~-360~~

$$C: \begin{array}{r} -100 \\ +1000 \\ \hline -0 \\ +0 \\ -0 \\ \hline 900 \\ \hline \end{array}$$

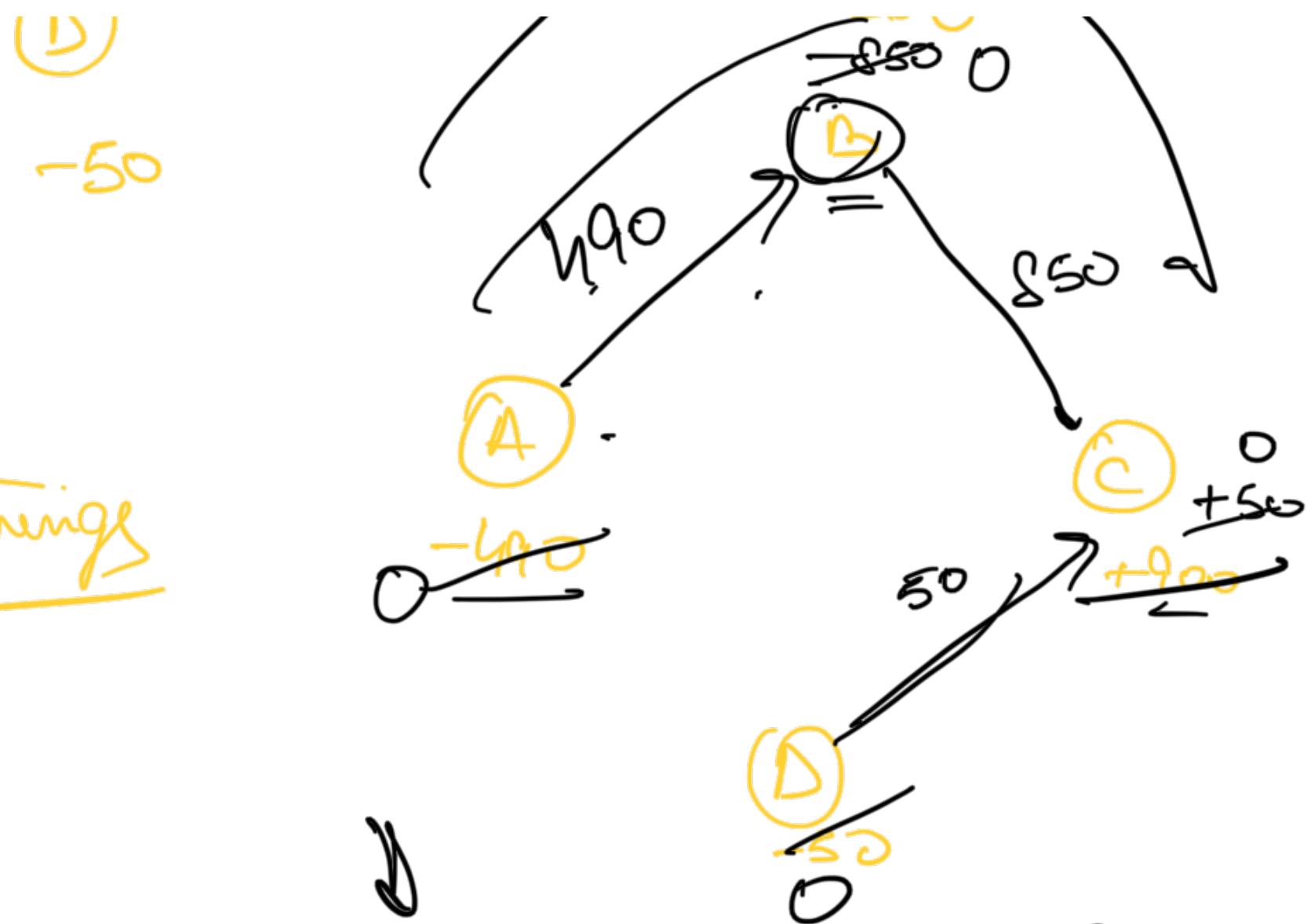
~~$$D: \begin{array}{r} -100 \\ +50 \\ \hline -50 \\ \hline \end{array}$$~~

$-850 \rightarrow 50$

$$\begin{array}{r} +900 \\ -50 \\ \hline +0 \\ \hline \end{array}$$



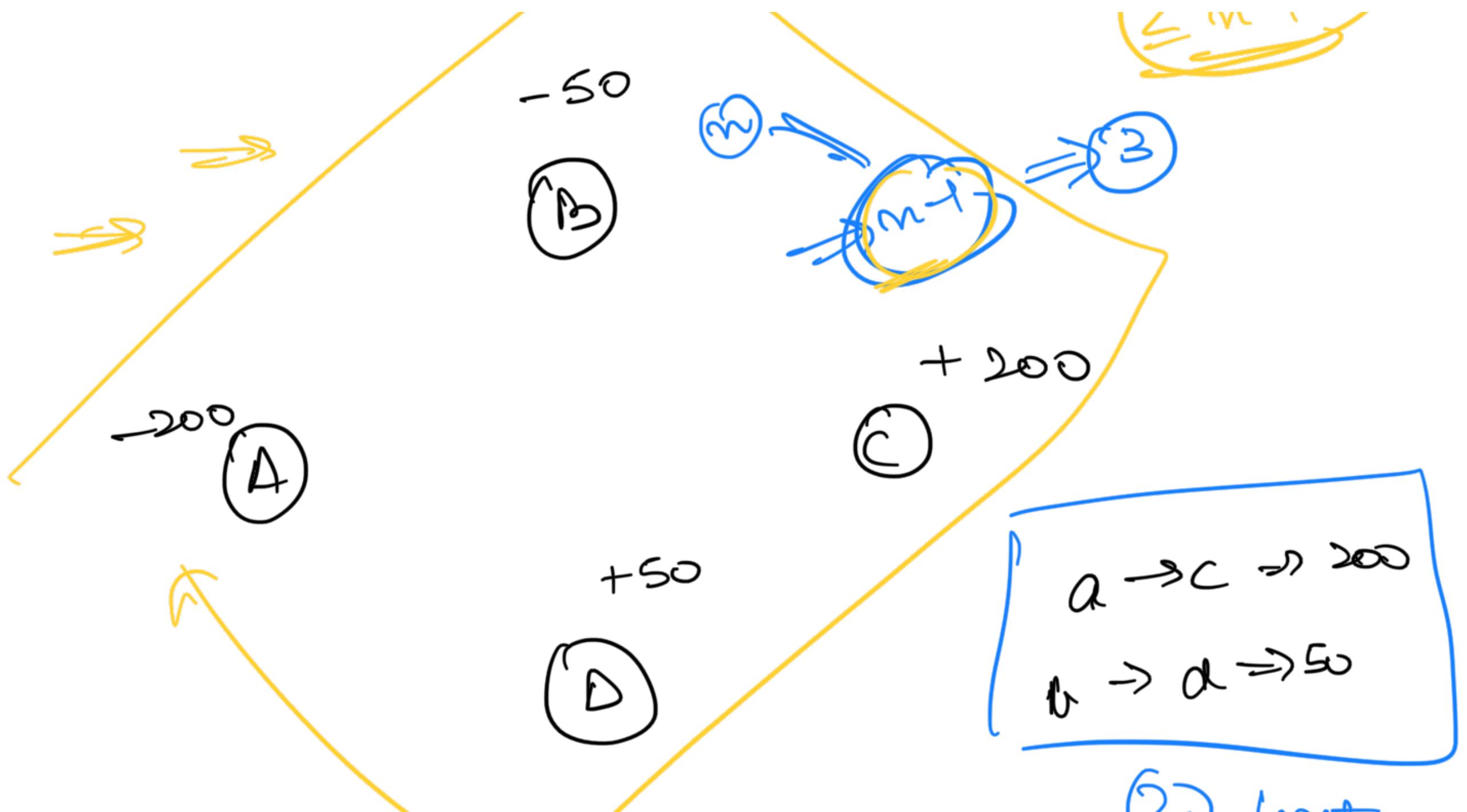
A bad things



$$A \rightarrow B \Rightarrow 500$$

$$B \rightarrow C \Rightarrow 500$$

$$D \rightarrow C \rightarrow 50$$



```

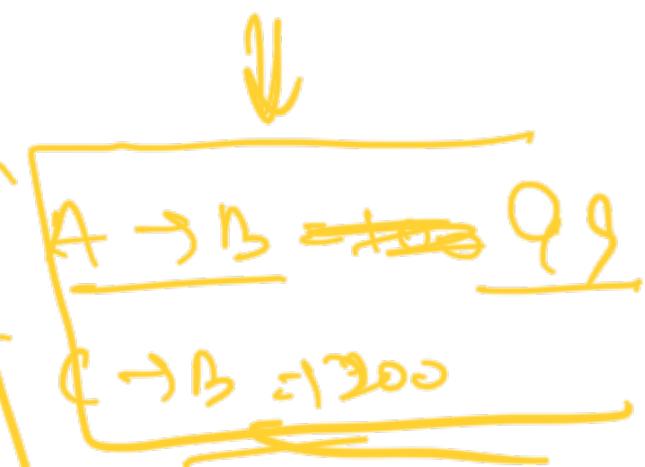
list <Transaction> SettleUp (List <expenses>)
  {
  }
  
```



-1

Total # of users across expenses

}



An algo to minimize the # of transaction



↓

$n \cdot m$

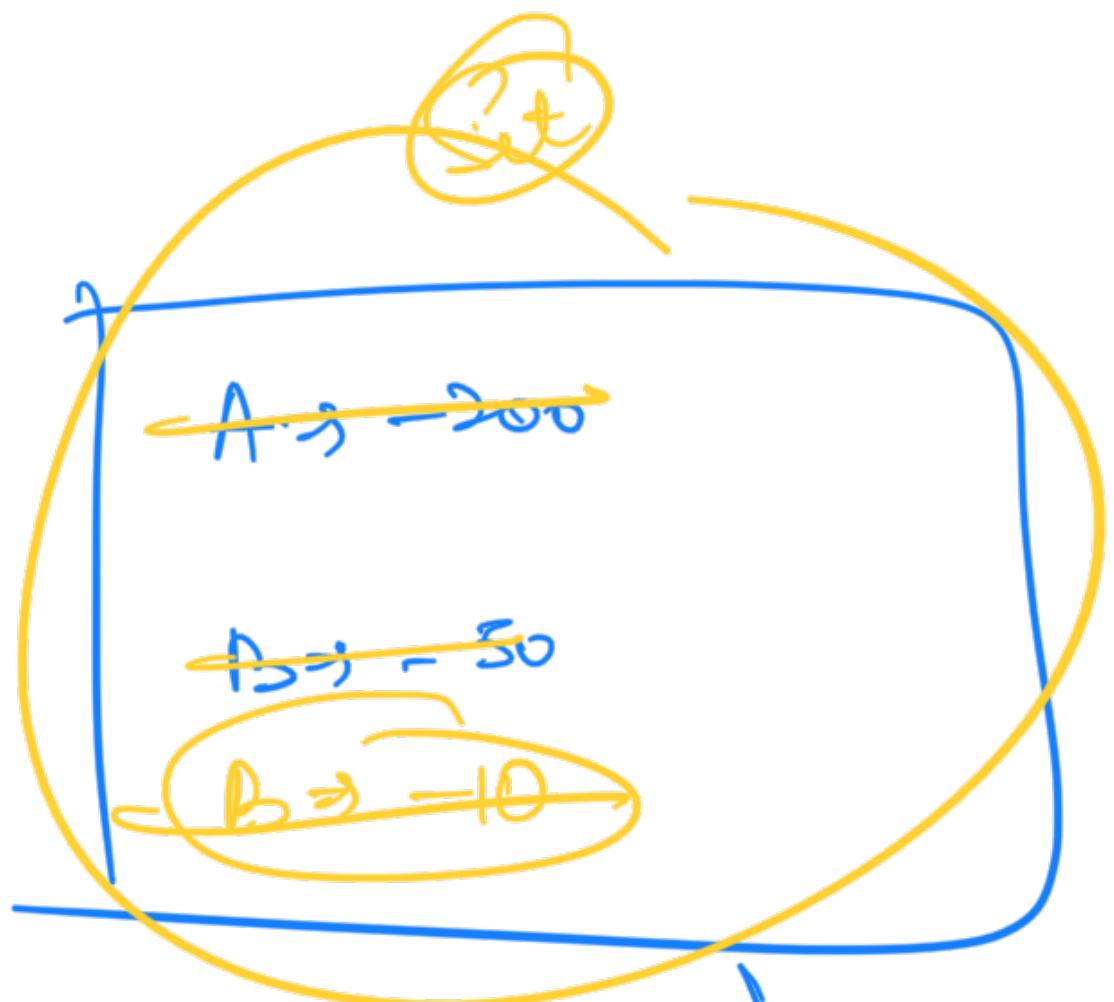
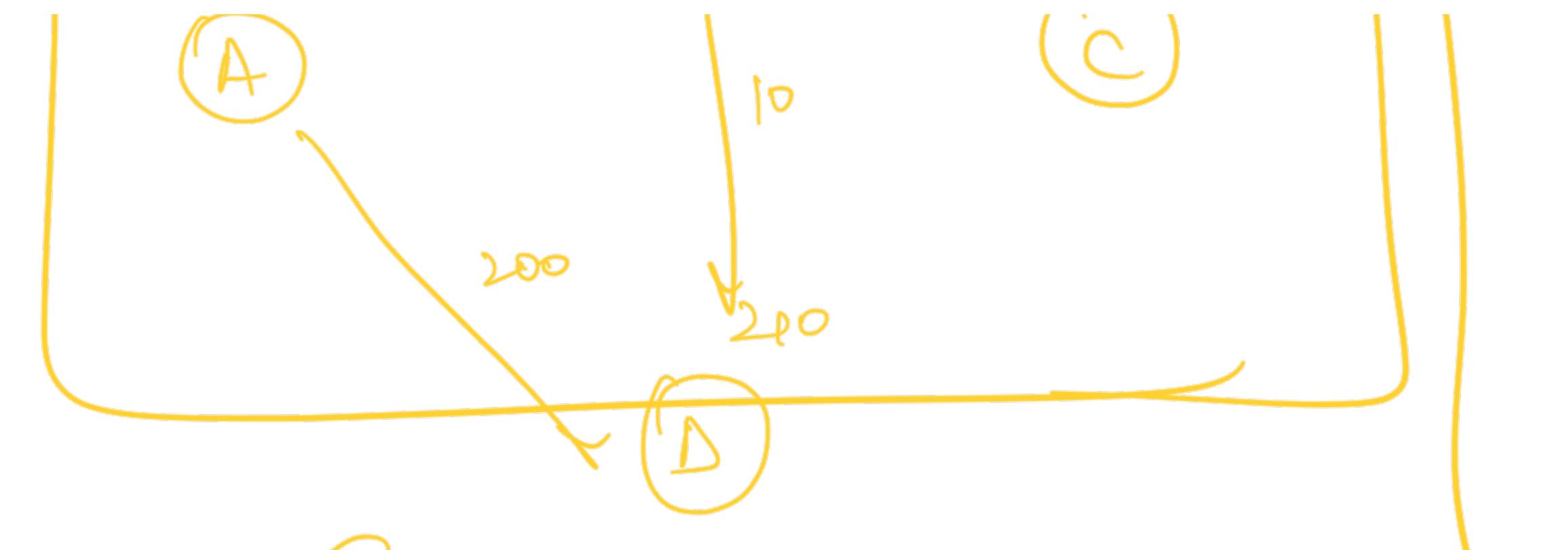
⇒

No polynomial Time
 $n \cdot m$



{ any NP hard problem can be converted to
 any other NP hard problem





Sel-

~~$C = 40$~~

~~$D = 240$~~

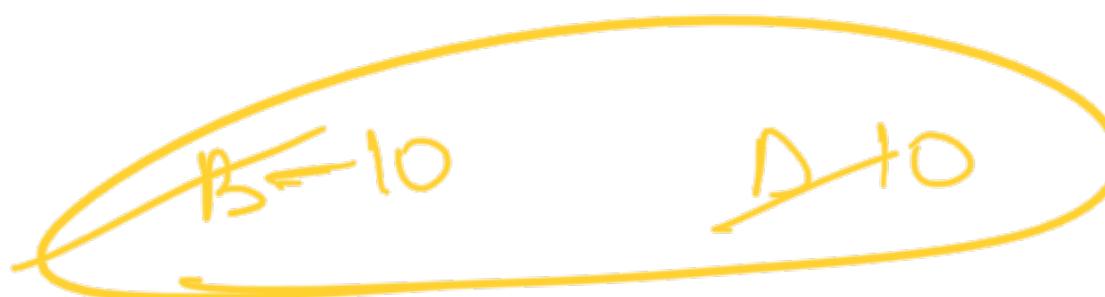
~~$D = 10$~~



$$B \rightarrow C = 40$$



$$A \rightarrow D = 200$$



$$B \rightarrow D = 10$$





$$(A \Rightarrow \underline{200} \quad D \Rightarrow \underline{210})$$

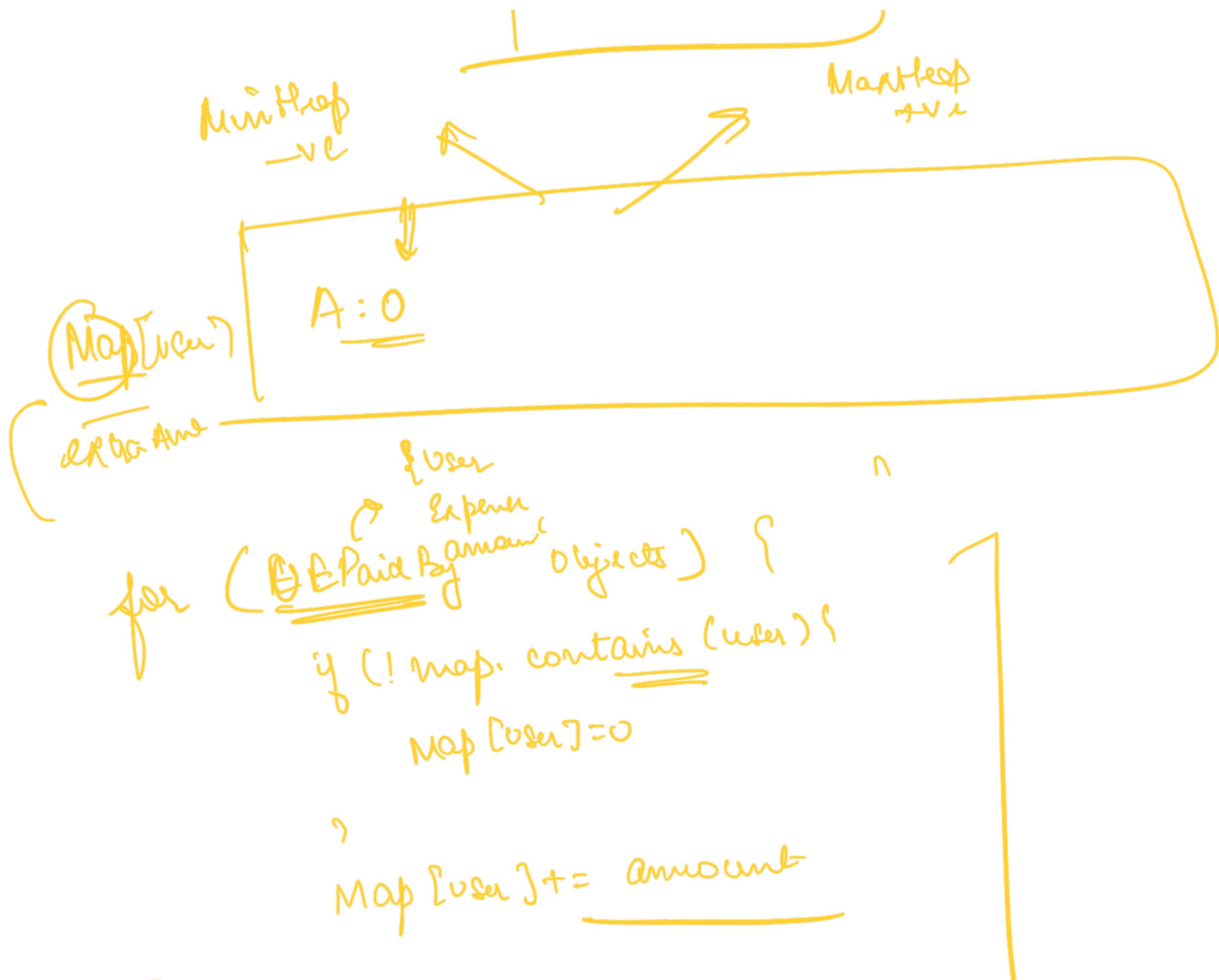
$A \rightarrow D \Rightarrow 200$

$$D \Rightarrow \underline{50}$$
$$C \Rightarrow \underline{60}$$

$$T B \rightarrow C = 40$$

$$B \Rightarrow \underline{10} \quad D \Rightarrow \underline{10}$$

$T B \rightarrow D \Rightarrow 10'$



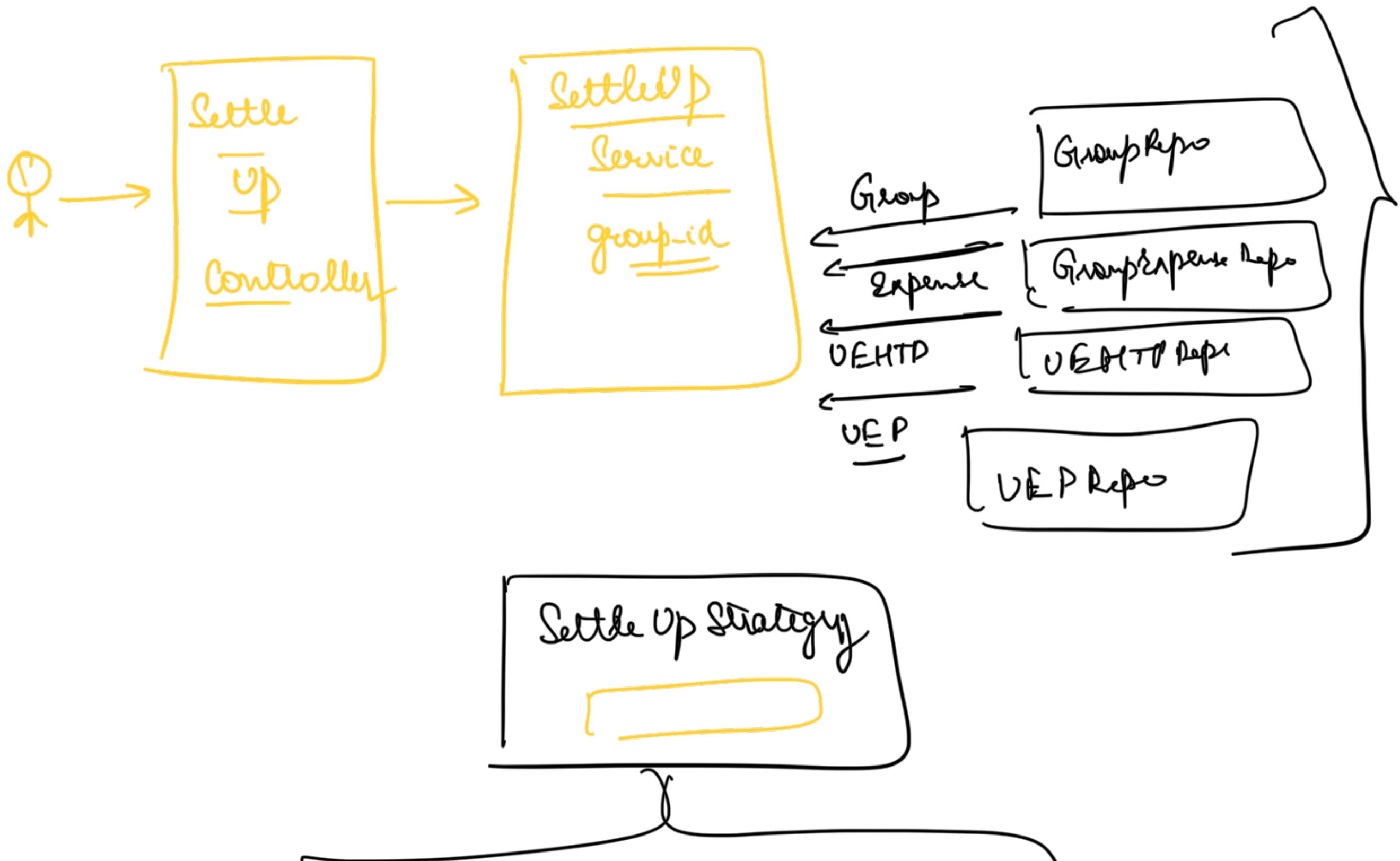
for (UEHTP objects) {
 if (!map.contains()) {
 map[use] = 0
 }
 map[use] += amount
}

X → C
(-40)

D
(50)
10

C → D → 40

/settle up | group ? group-id = " "



Give to Next User
Strategy

Max Max Sett Up
Strategy

Break till 10:44 PM

Learning Plan (Group_id user_to_learn user_id)

removing - user isn't

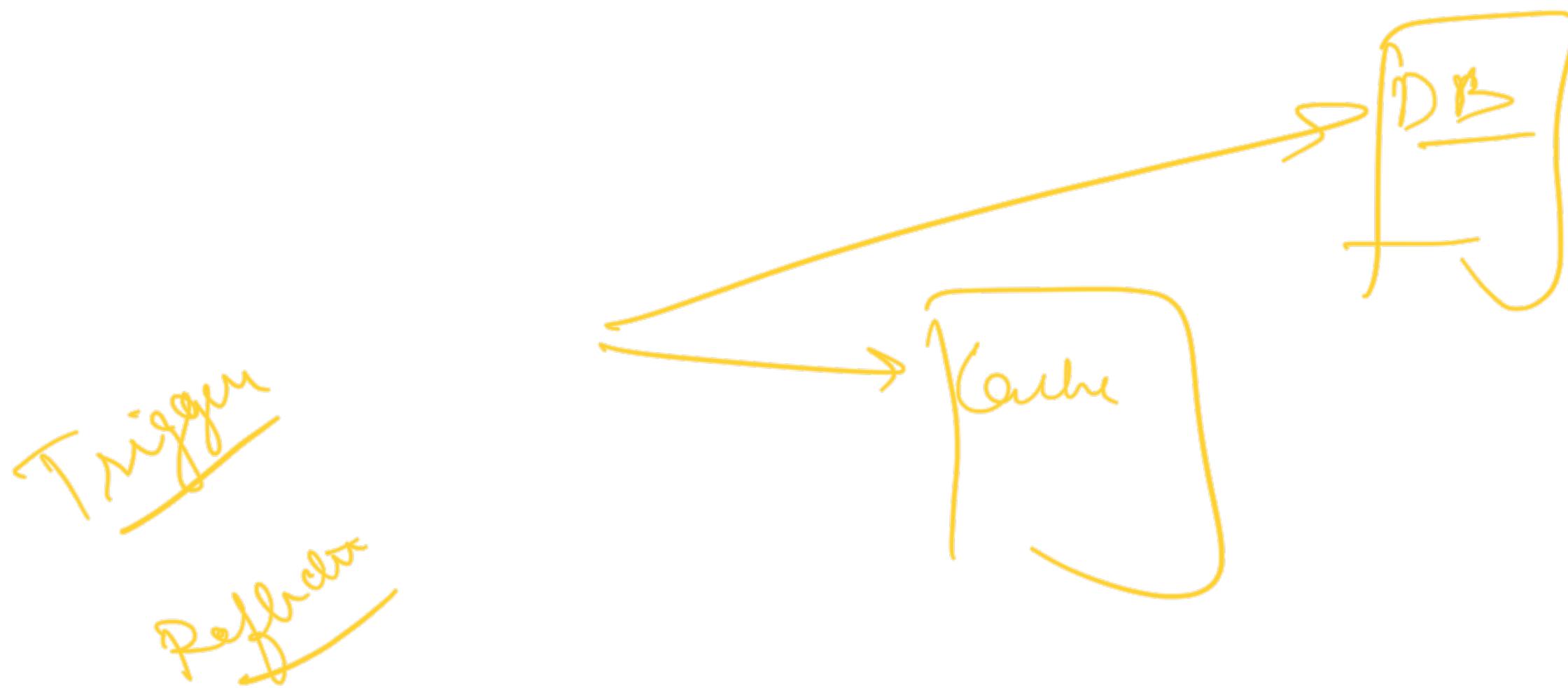
- get group details from group rep
- Check if removing - user is admin
- Yes {

} No }

throw exception

}

Scalene - way in structor - dathus



Code Splitting

- ① Register
- ② Login
- ③ Settle Up