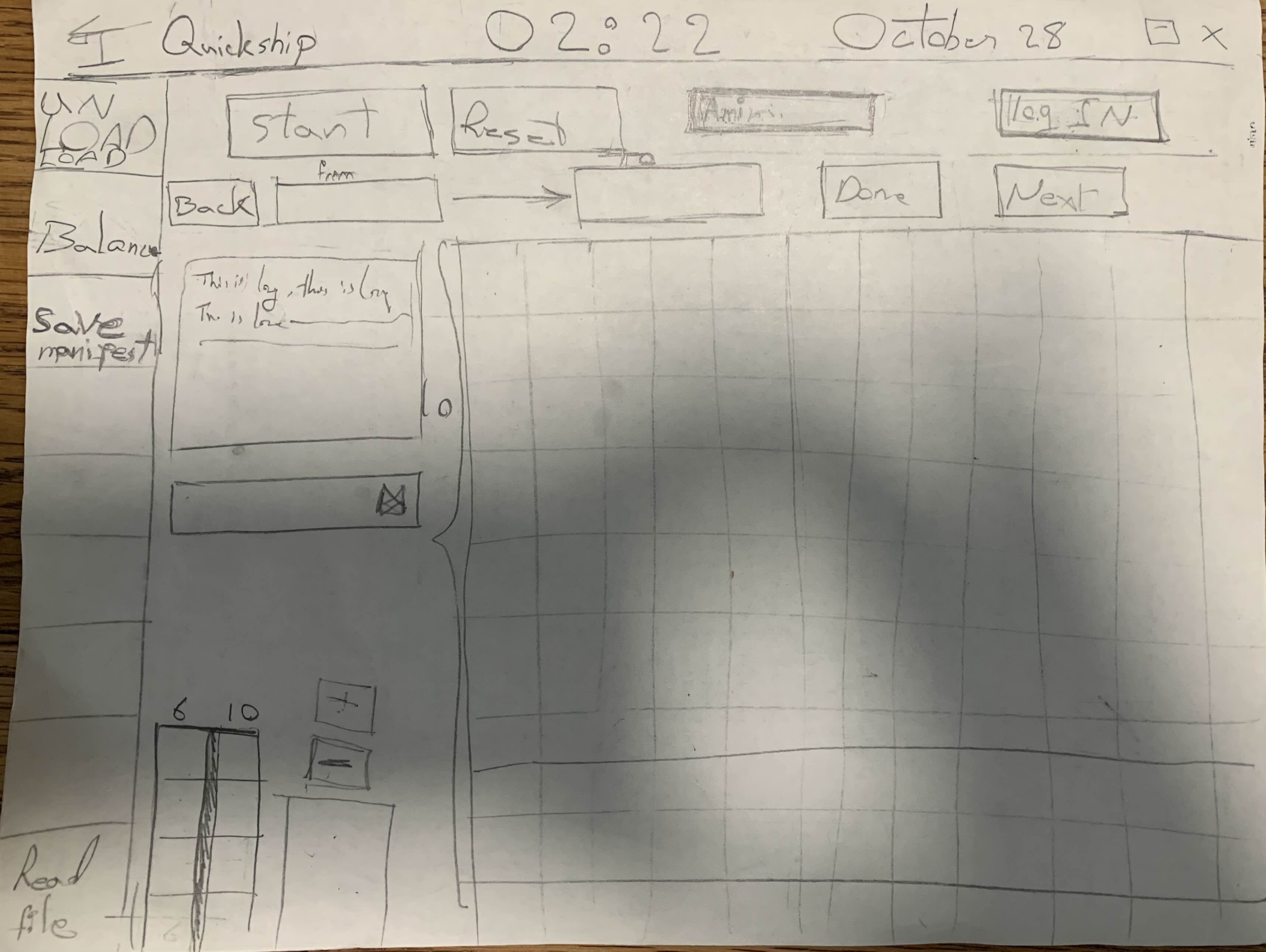
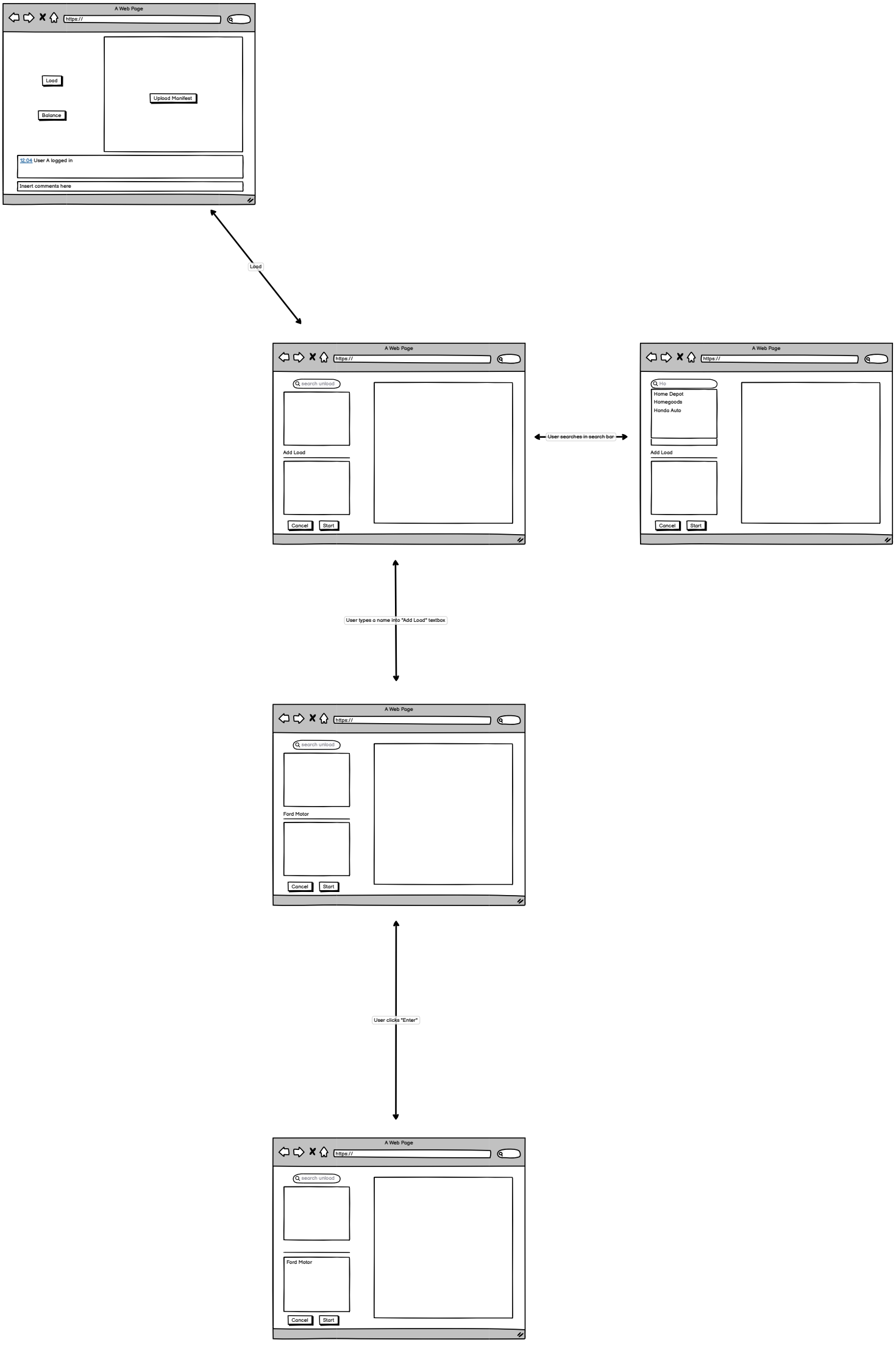
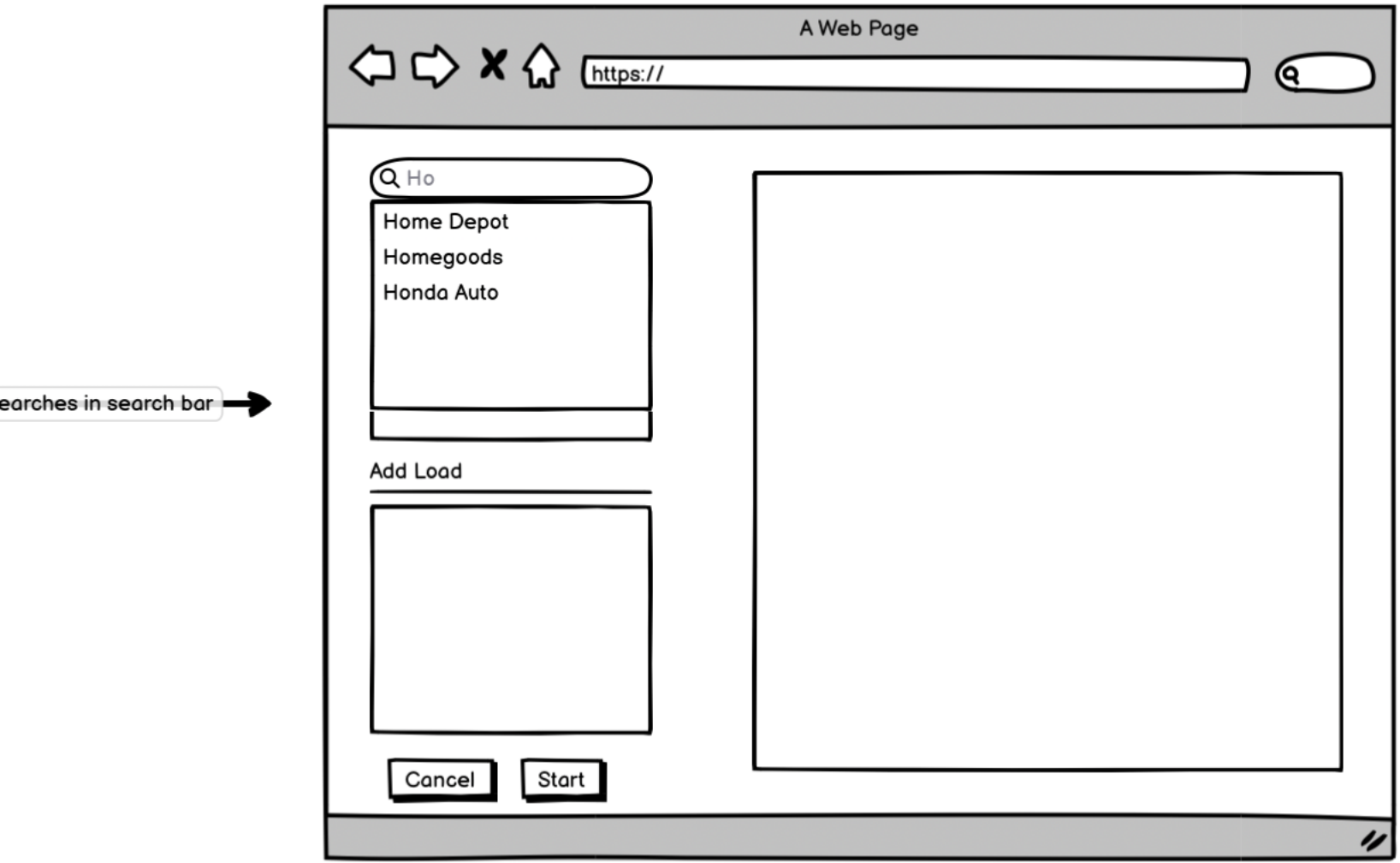
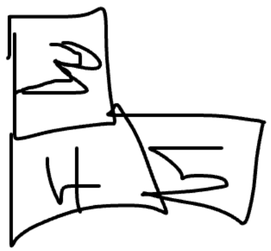

Quickship
02:22
October 28
UNLOAD
start
Reset
Animi
log IN
from
Back
Done
Next
Balance
This is log, this is log
This is log
Save manifest
10
6
10
+
=
Read file

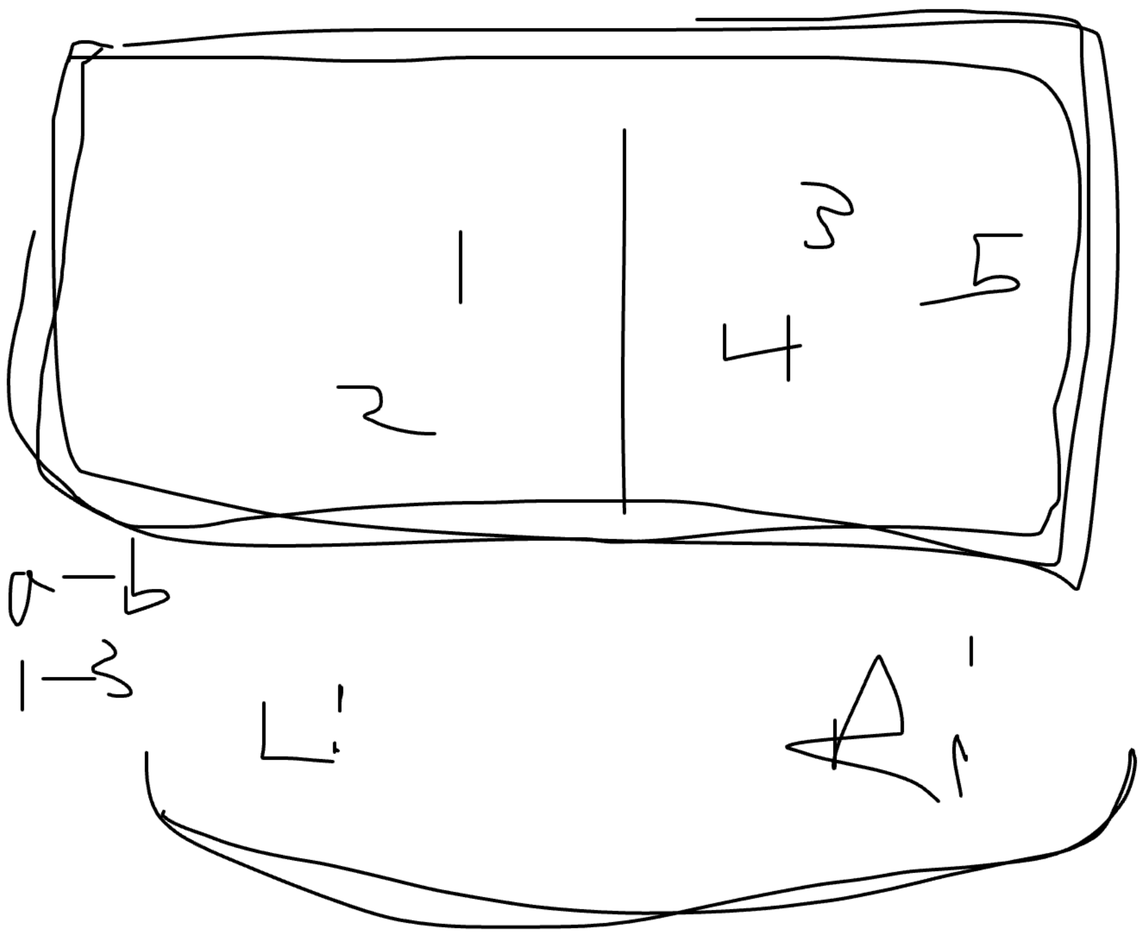

A Web Page
https://
Load
Balance
Upload Manifest
12:04 User A logged in
Insert comments here
Load
A Web Page
https://
search unload
Add Load
Cancel
Start
User searches in search bar
A Web Page
https://
Ho
Home Depot
Homegoods
Honda Auto
Add Load
Cancel
Start
User types a name into "Add Load" textbox
A Web Page
https://
search unload
Ford Motor
Cancel
Start
User clicks "Enter"
A Web Page
https://
search unload
Ford Motor
Cancel
Start


A Web Page
https://
Ho
Home Depot
Homegoods
Honda Auto
earches in search bar
Add Load
Cancel
Start

a-b



a-b



a-b

1-3

$$F(i,k) = \begin{cases} 0, & i = 1 \;\&\; x_1 = k \\ \infty, & i = 1 \;\&\; x \neq k \\ \min \begin{pmatrix} F(i-1, k - x_i) + c_L(x_i) \\ F(i-1, k) + c_R(x_i) \end{pmatrix} \end{cases}$$

Container

Var:
name
weight

CostOfCraneToContainer

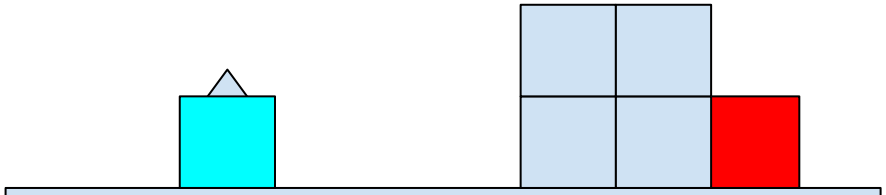
Loop all possible moves

Ship

Var:
Grid[][]
TopContainer[] // size is the number of columns
Crane coords (x,y)

Func:
Ship x;
x.Grid = convert ManifestGrid
x.Grid = y

FindTopContainers()



ManifestGridConverter

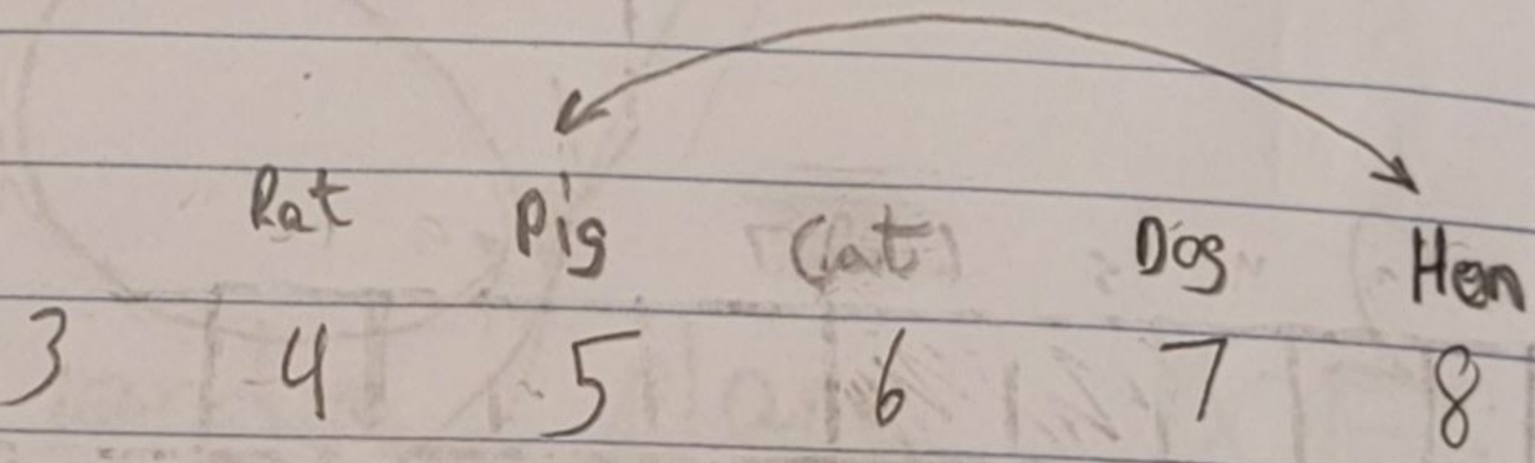
Func:

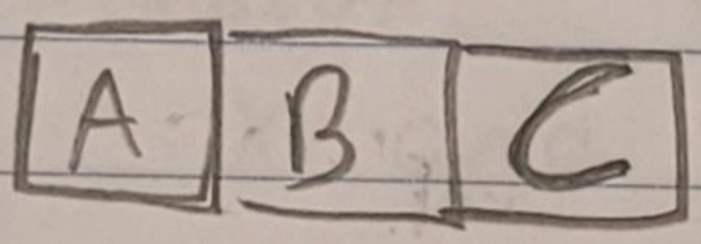
```
ConvertManifestToGrid(manifest)
{
        //Add content to grid
        Return grid
}

ConvertGridToManifest(manifest)
{
        //Writes to manifest using grid contents
}
```

Balance

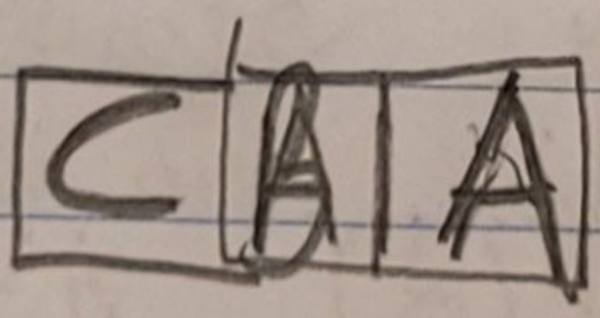
Var: orderOfOp[]
estimateTime

Cat	dog	Pig	Hen	Rat		
2	3	4	5	6	7	8

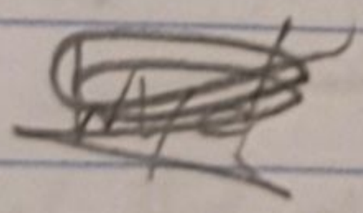




4 + 0 + 3 + 2



2



yes in break
how about row
At start

Rs Hen

Cat	dog	Pig	Hen	Rat		
2	3	4	5	6	7	8

	Rat	Pig	(Cat)	Dog	Hen
2	3	4	5	6	7	8

A	B	C

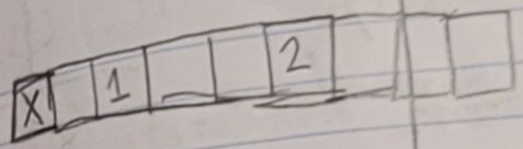
4+4+0+3+2

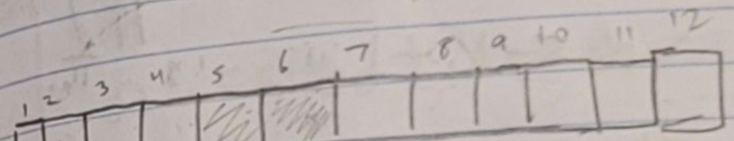
C	B	A

2

yes in brak
how about now
A+ start

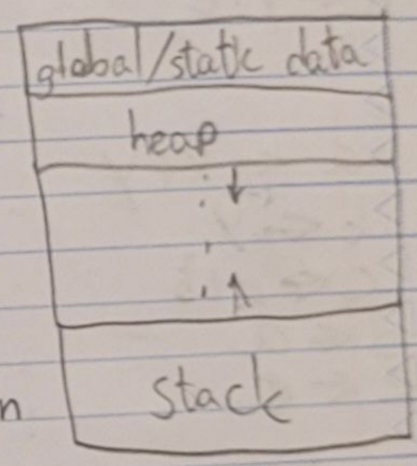
Big Iter





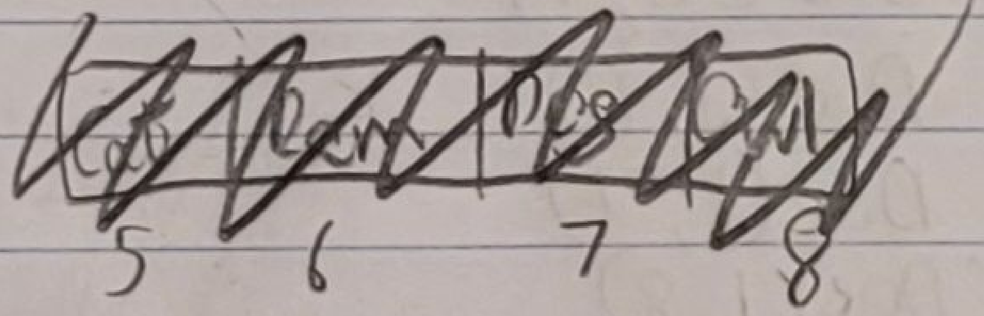
p

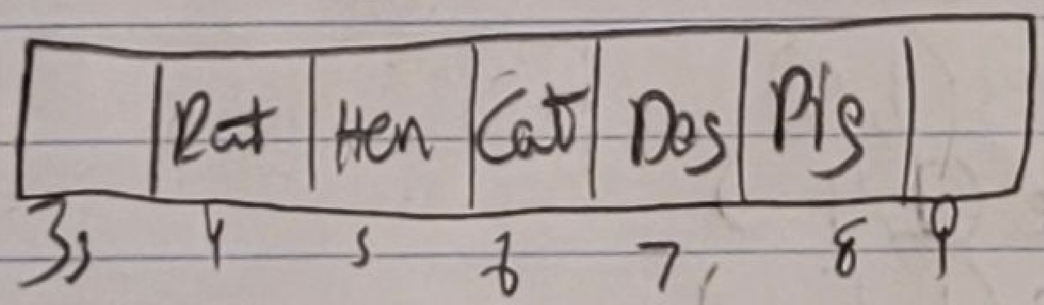
p[0], ... p[n-1]

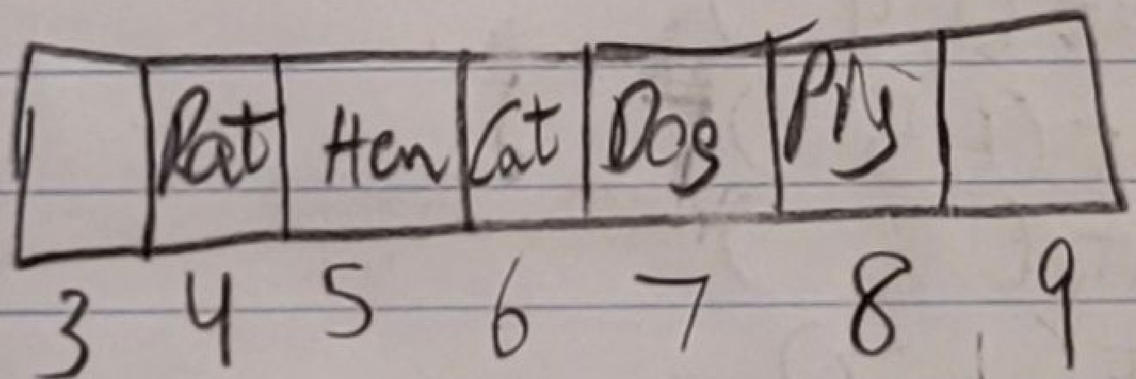


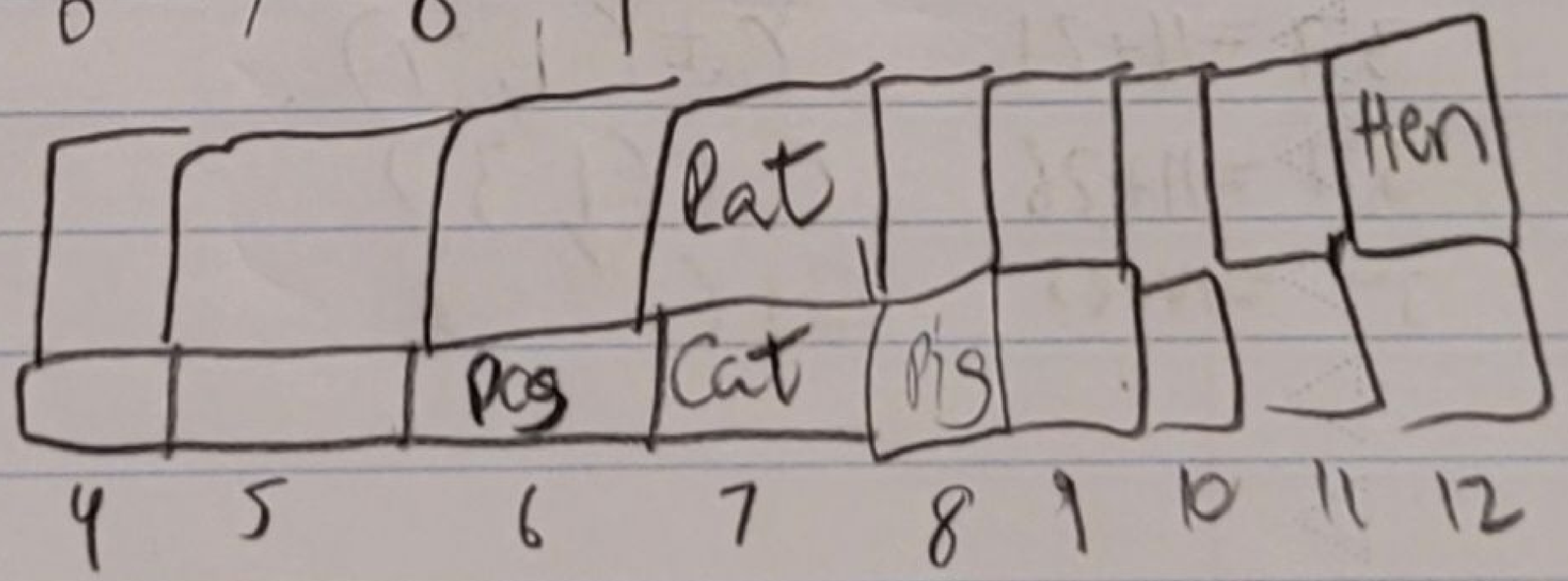
i, a, a[0], ... a[9], n

base pointer value (now) pushed to stack
base pointer value set to stack pointer value (top)
Memory allocated for fo
a put on stack and set to 3
Argument a passed into mem via register
the called
memory deallocated for the fop
base pointer set to what
pop old address of
ret a


5
6
7
8


Rat
Hen
Cat
Dog
Pig
3
4
5
6
7
8
9


Rat
Hen
Cat
Dog
Pig
3
4
5
6
7
8
9


Rat
Hen
Dog
Cat
Pig
4
5
6
7
8
9
10
11
12

$+1+4+6$

24+3+12=39

10×39

(36) 15+23=(38)

27
+12
39

1 2 3 4 5 6

1+6

5

Rat	Dos	Eme	Cat	Cow	Doe
4	5	6	7	8	9

0+2+5+ 44 +5+7 = 7+44+12 = 7+66=(73)

23 24 25 26 27

7+6+7+5+2

13+12+2

25+2

9

(26)

(26)

38 = 9+28 Scat

38 = 9+27 Doe (2,4)

36 = 13+23 Doe (1,6)

36 = 14+22 Doe (1,7)

36 = 15+21 Doe (1,8)

36 = 16+20 Does (1,9)

36 = 10+26 Dee (1,4)

36 = 12+24 Doe (1,5)

37 = 12+25 Cat (1,4)

37 = 15+22 Cat (1,5)

37 = 9+28 Cat (

37 = 16+21 Cat (1,7)

37 = 11+26 Cat (1,3)

37 = 14+23 Cat (

currState.operationList is undefined

	g	h
(1,4) → (1,7)	14	3
(1,4) → (1,8)	15	3

Case 1

Move crane to (1,3) and move container to (1,7)
(Estimate 14 minutes)

Move crane back to starting location at (1,9)

- Consider minimize

Case 3 falls

1	2	3
4	5	6
7		8

Looking for
(1,9) → (1,6)
(Owl) 3 to left

(3,1) → (1,23)

21+11 = 32

9+2+9

4+4+2

4+4+2

8+32-1 = 39

after

(1,5) → (1,6)

20+36-6 = 50

+1

8+28+4 = 12+28 = 40+9
= 49

4+32+8 = 44+4+2
+4 +2 = 50

3+4+2

8+(32-5)+4

12+8+(32-6)+4

Then initially after (1,5) on ship
moves to (1,6) on buffer

9+

4+(28-6) = 22+4 = 26

8+(32-6)+
= 38

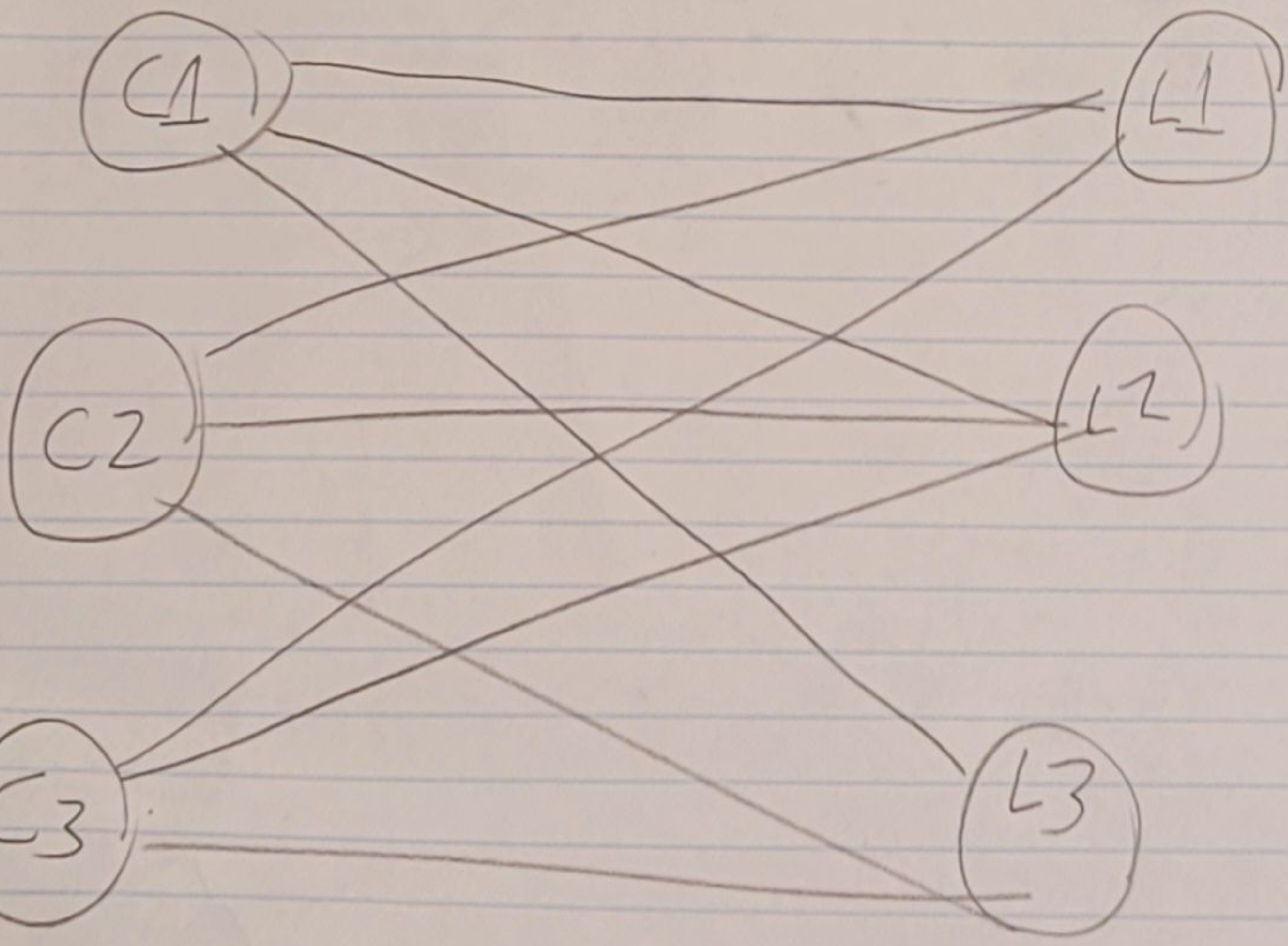
12

Make a commit to balance-algorithm

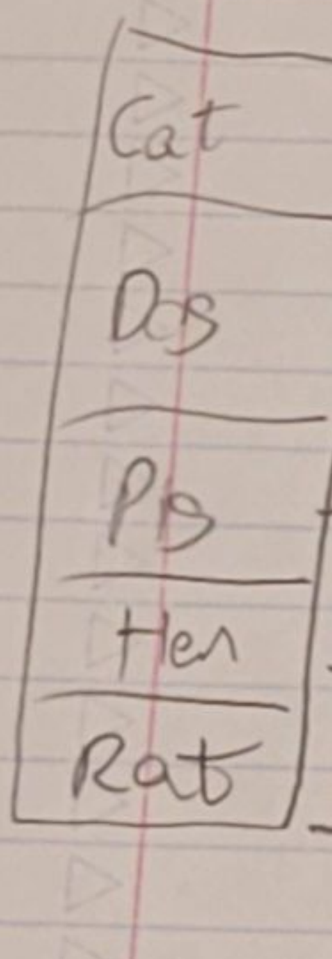
Contains everything it had before, PLUS

1) Balance Operation With Buffer
2) Balance Ship State With Buffer

gridList operationList goalState



96 Cat | 8 Dog | 4 Pig | 4 Hen | 1 Rat



→ (1,6)

→ (1,7)

→ (1,5) or (1,6)

→ (1,5) or (1,6)

→ (1,4)

(1,6)

[(1,6)]

[(1,7)]

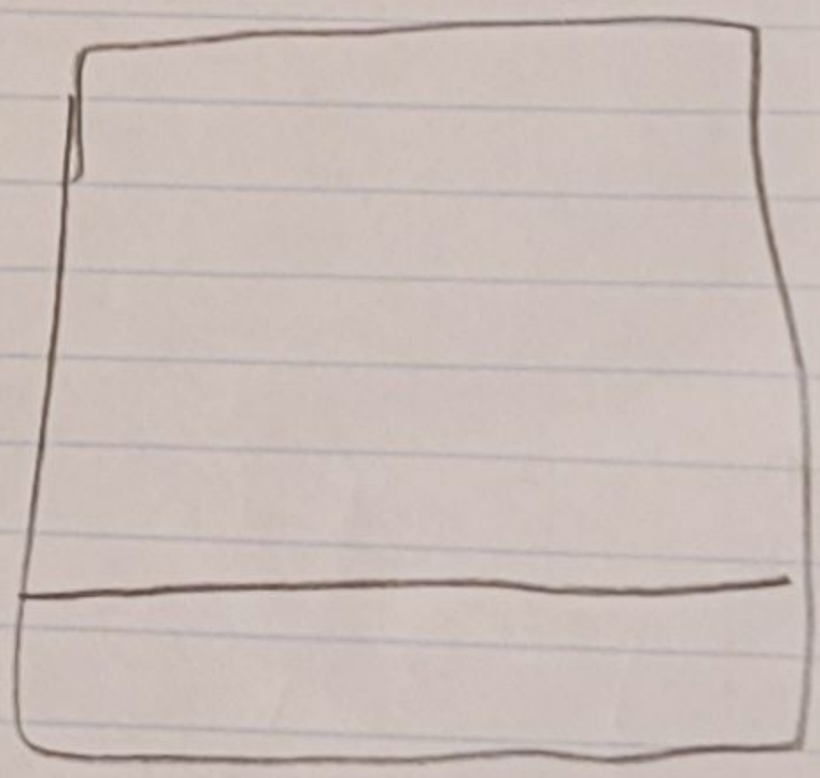
[(1,5),(1,6)]

(1,4)

0 → Cat

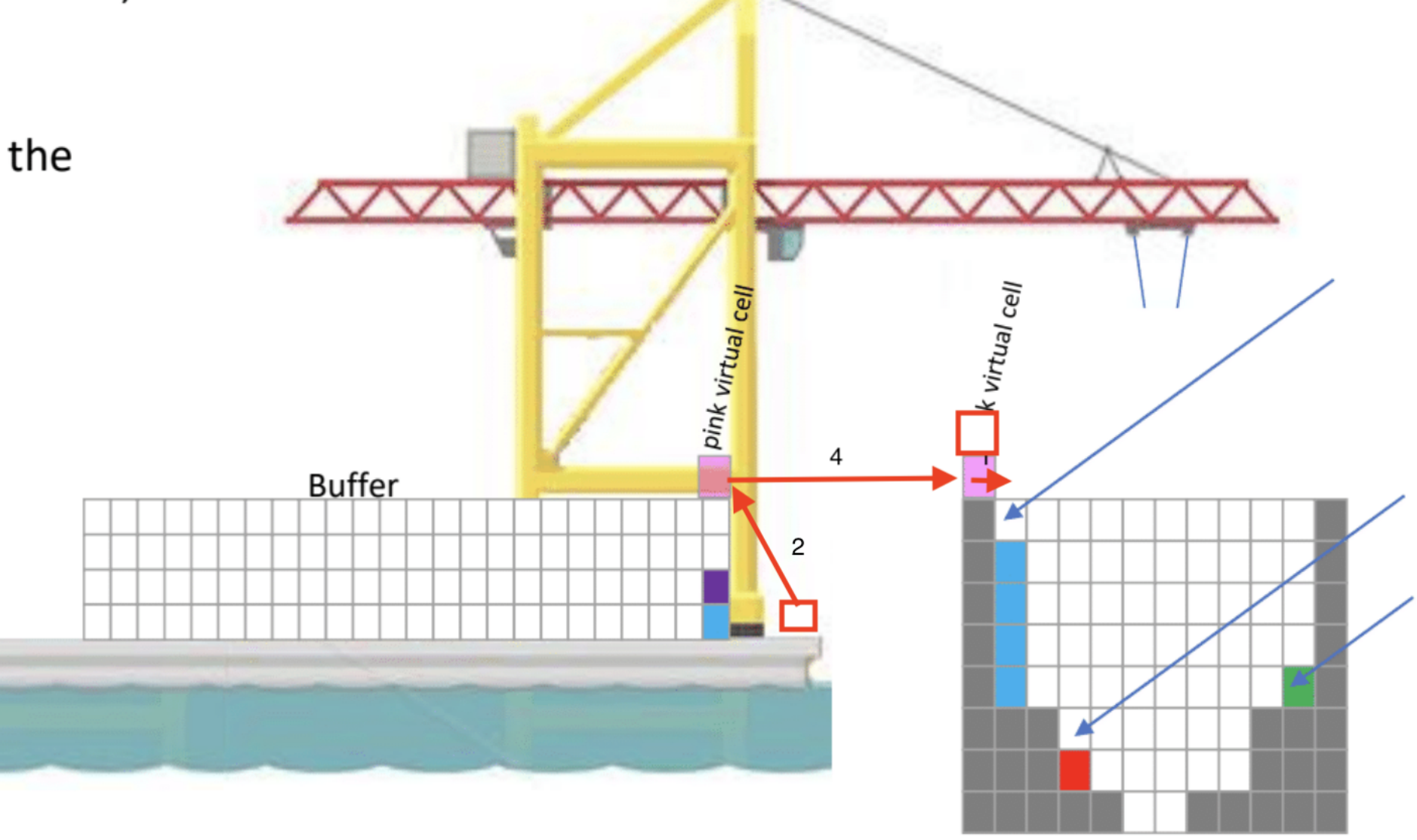
1 → Dog

2 → Pig → Hen

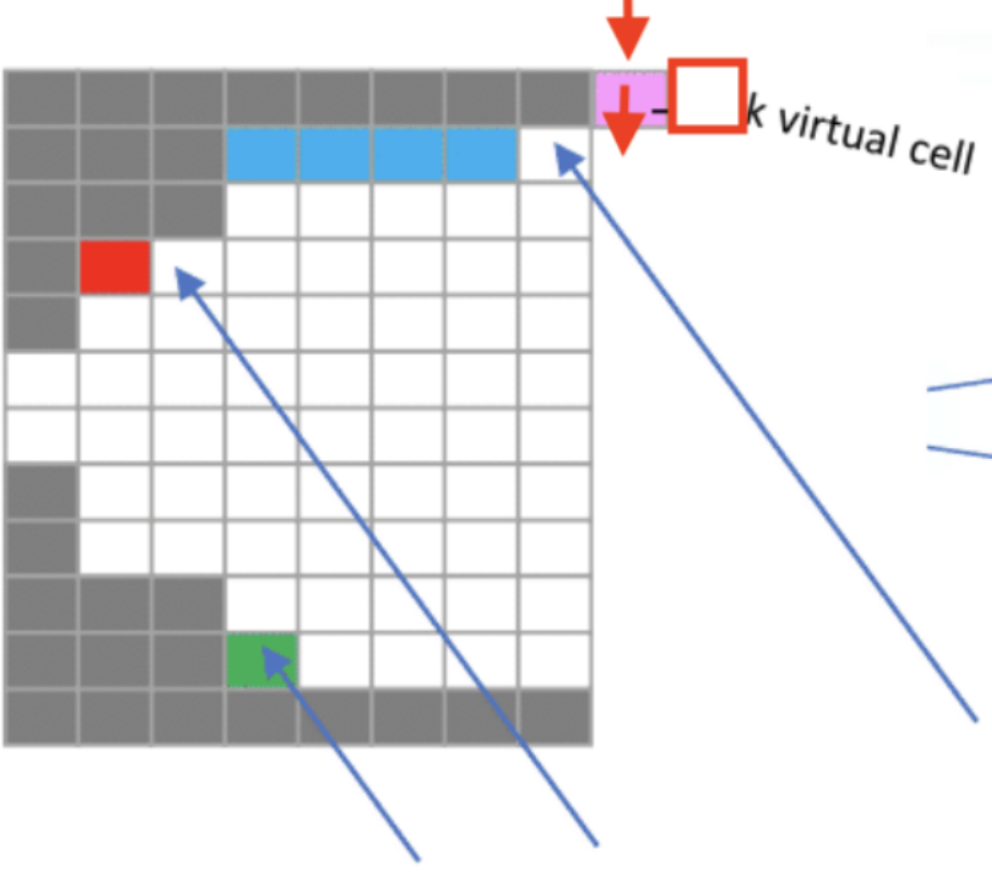


duplicatesList

the



ove
e three


k virtual cell

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39