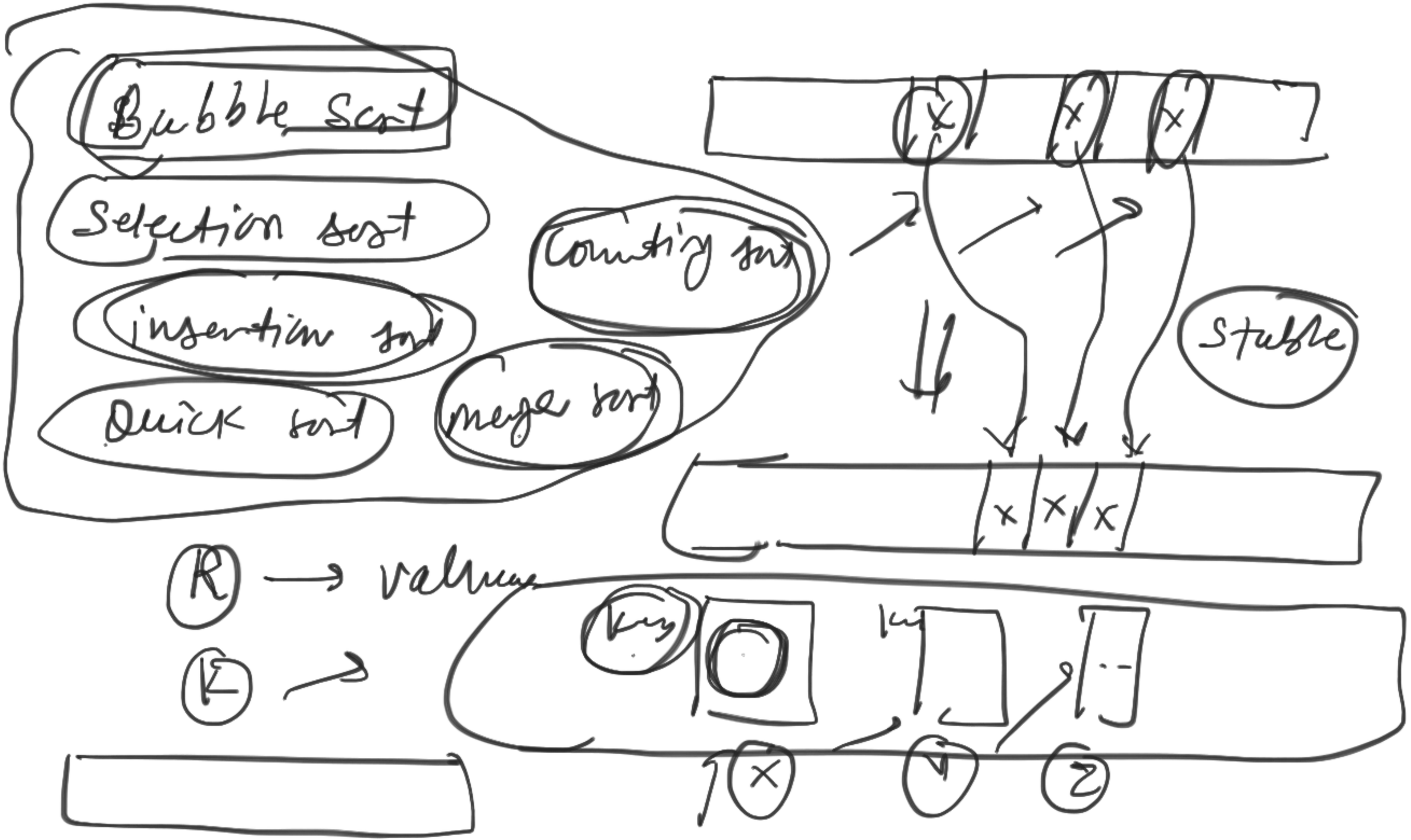
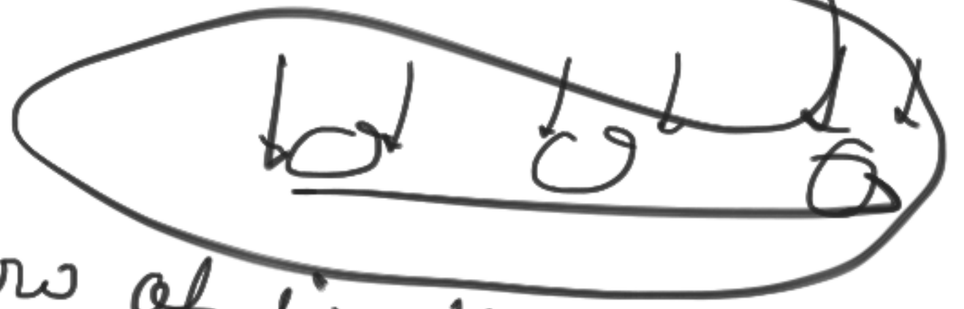

Bubble sort
Selection sort
insertion sort
Quick sort
Counting sort
merge sort
Stable
x
x
x
x
x
x
R → value
K →
key
key
X
Y
Z

Sliding window ⇒ window ⇒ variable

⇒ window ⇒ fixed

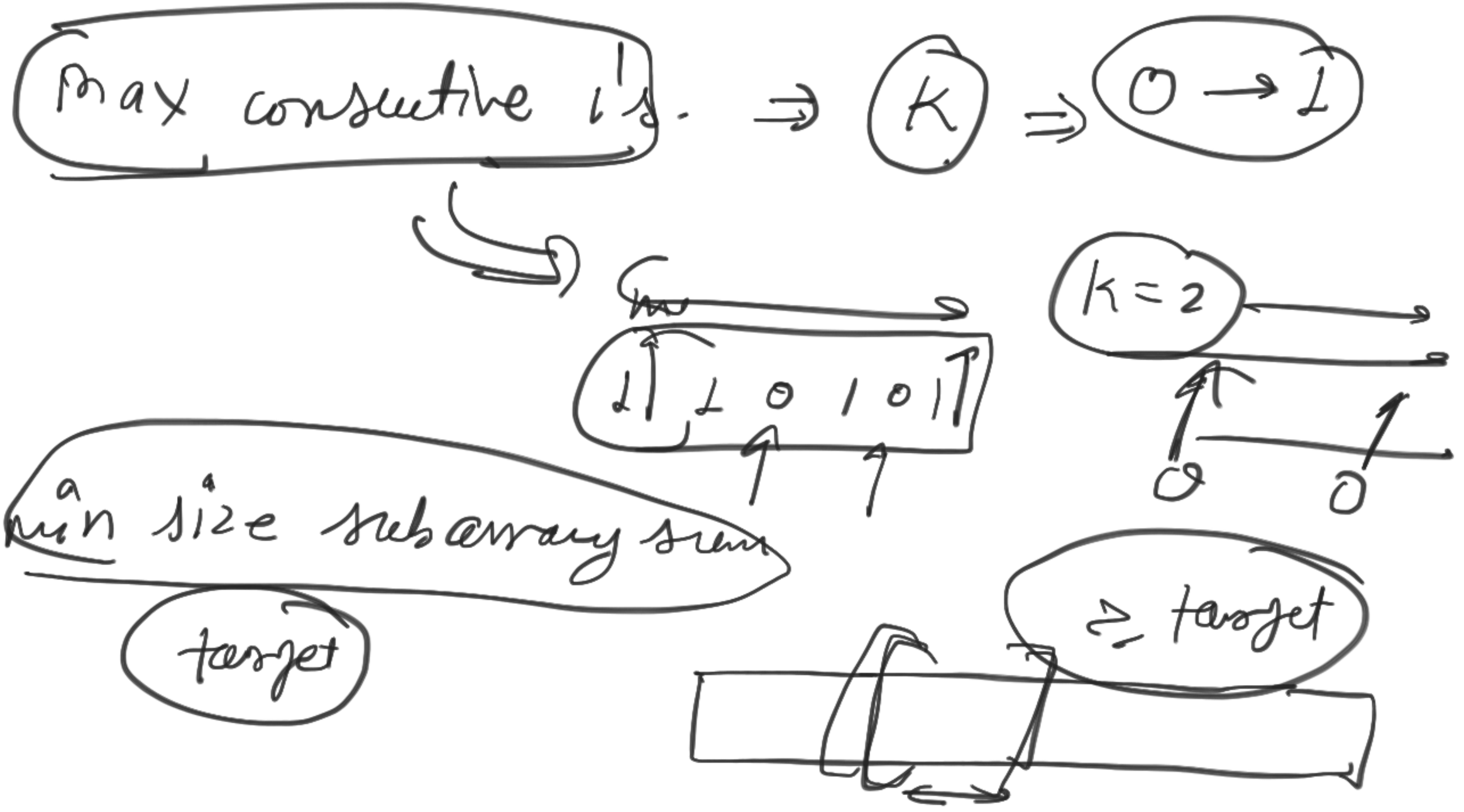


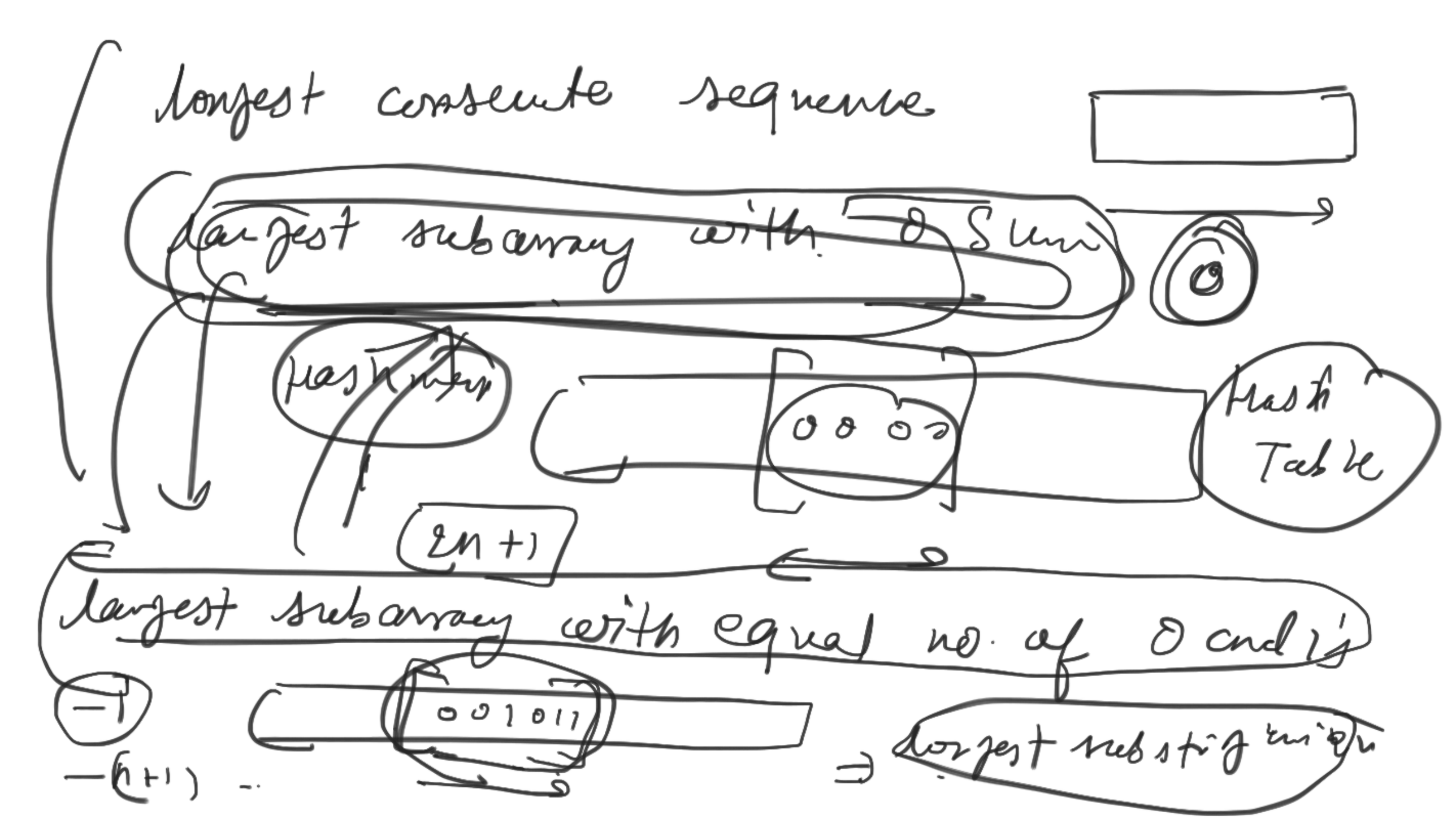
Count distinct element

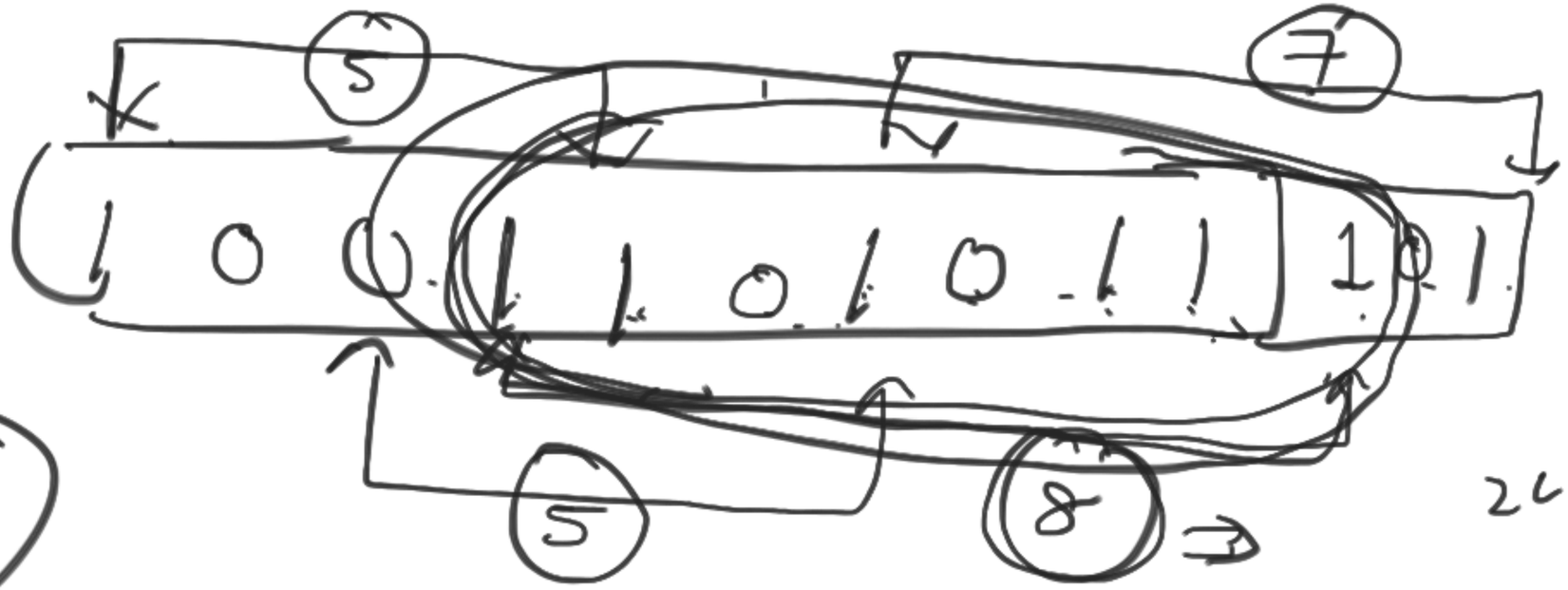
⇒ in every window of size k.

⇒ max of every window of size k

⇒ sliding window max


Max consuctive 1's.
⇒
K
⇒
0 → 1
1 1 0 1 0 1
k = 2
0 0
min size subarray sum
target
≥ target


longest consecute sequence
largest subarray with 0 Sum
0
Hash Map
0 0 0 0
Hash Table
2n+1
largest subarray with equal no. of 0 and 1
-1
0 0 1 0 1 1
-(n+1)
longest substring



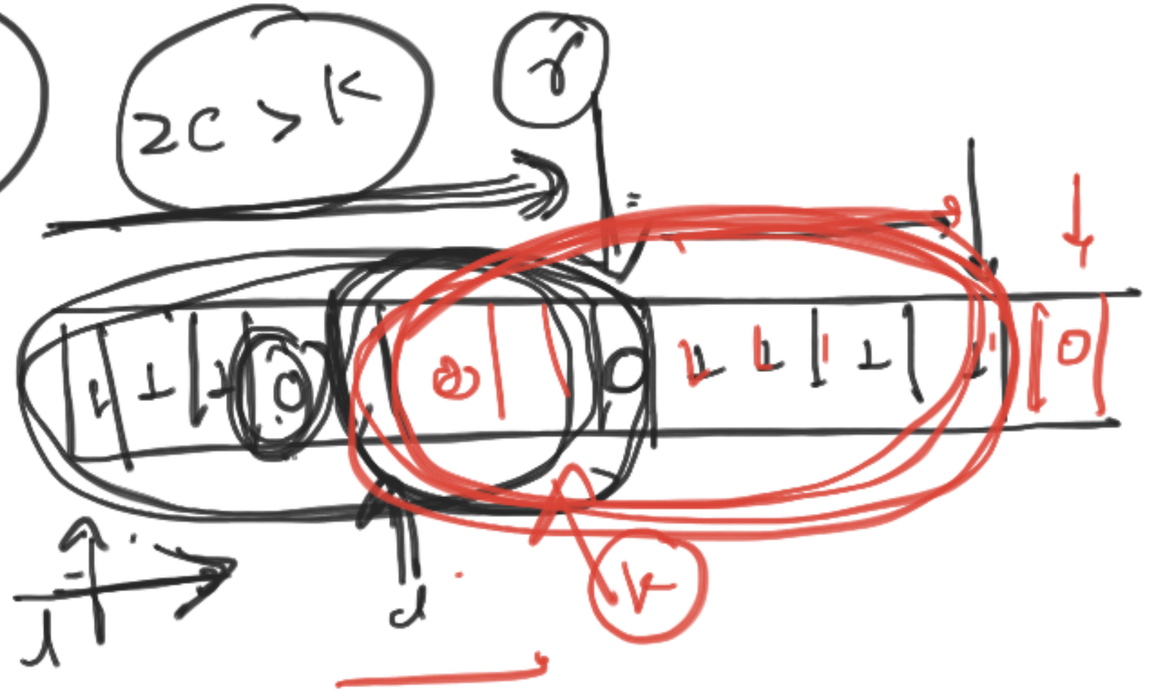
$K=2$

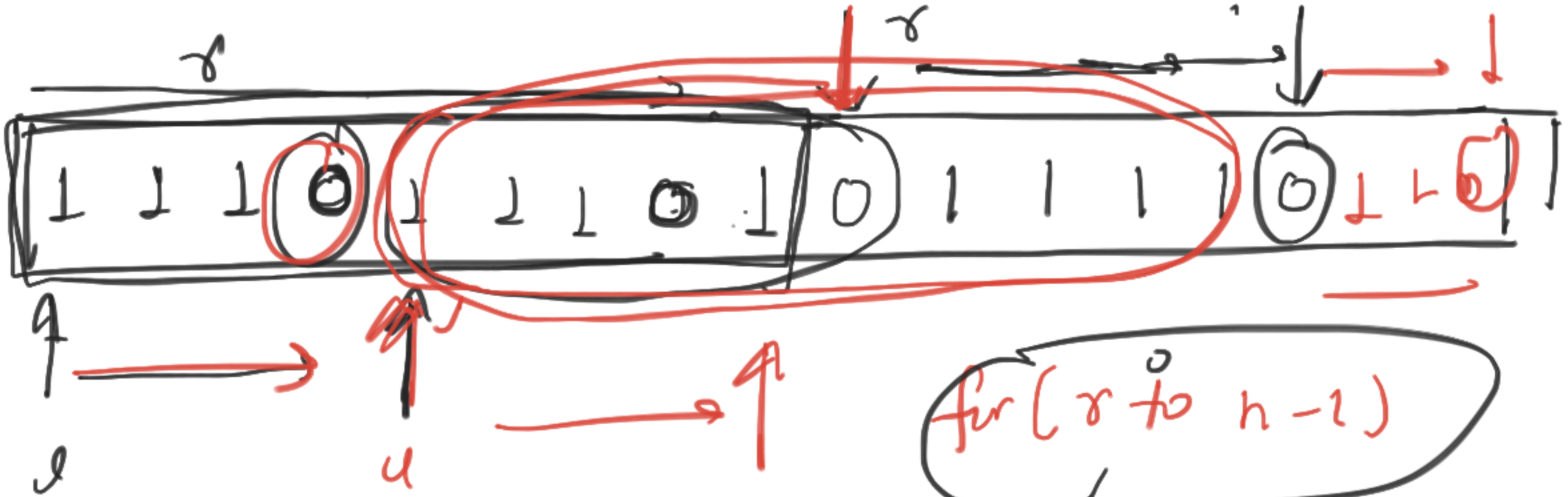
$2c < K$

$\leq K$

$> K$

$2c > K$





K = 2

maxOneCount ⇒

for(r to n-1)

{ if(A[r] == 0)

zc = zc + 1

if(zc > K)

{

}

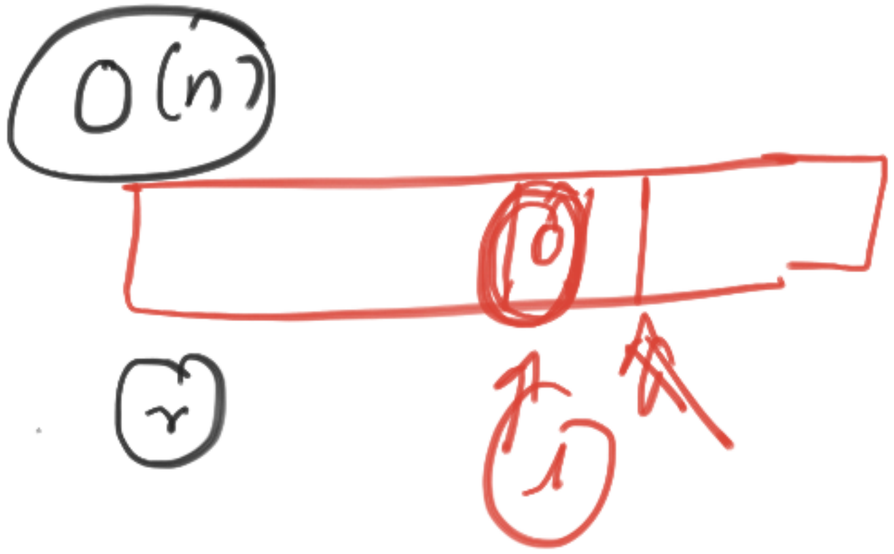
(while(x[l]==1)

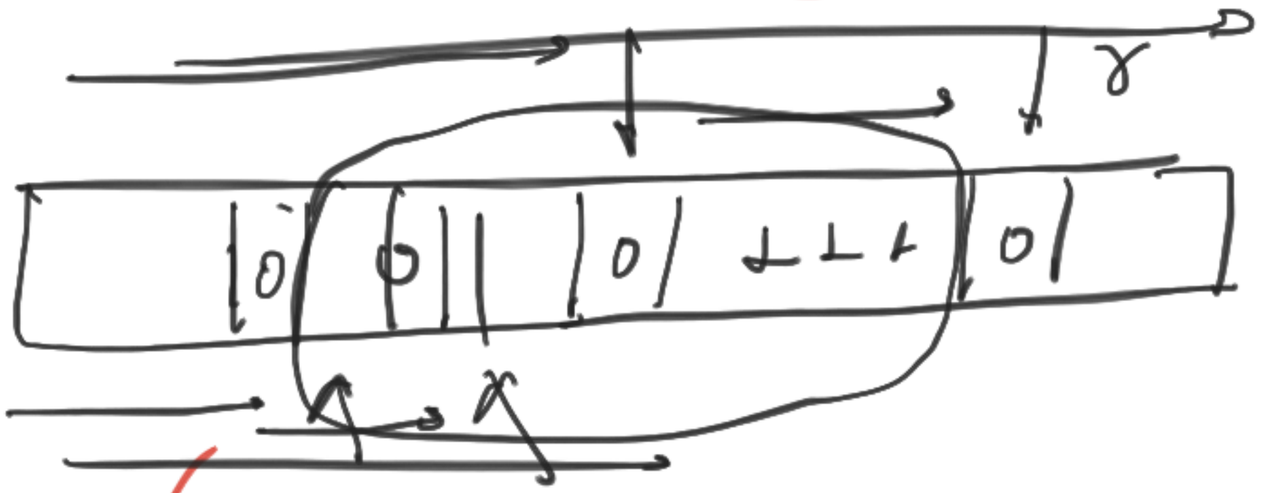
l = l+1

zc = zc-1

l = l+1

}

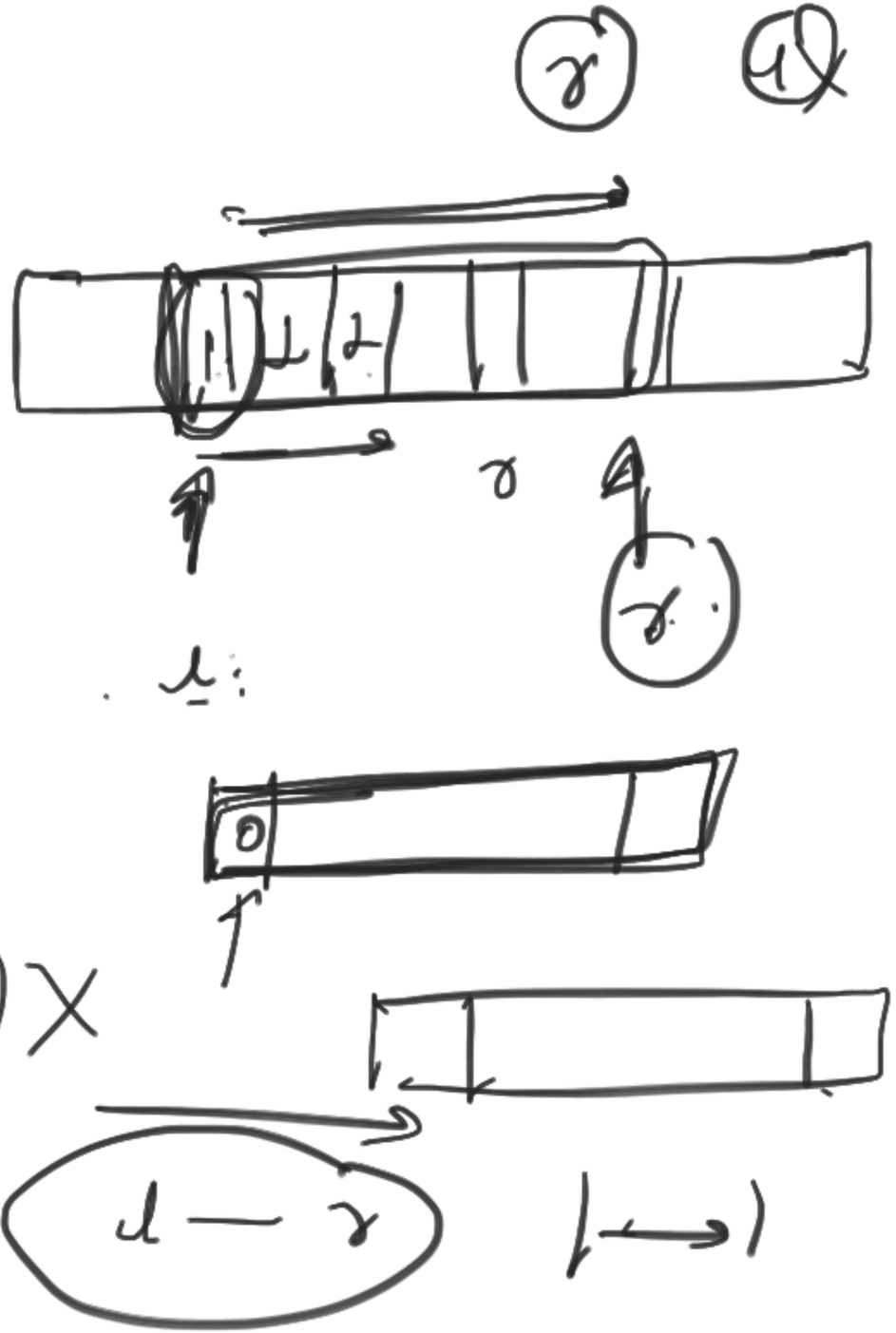




maxoneCount = max(maxoneCount, r-l+1)

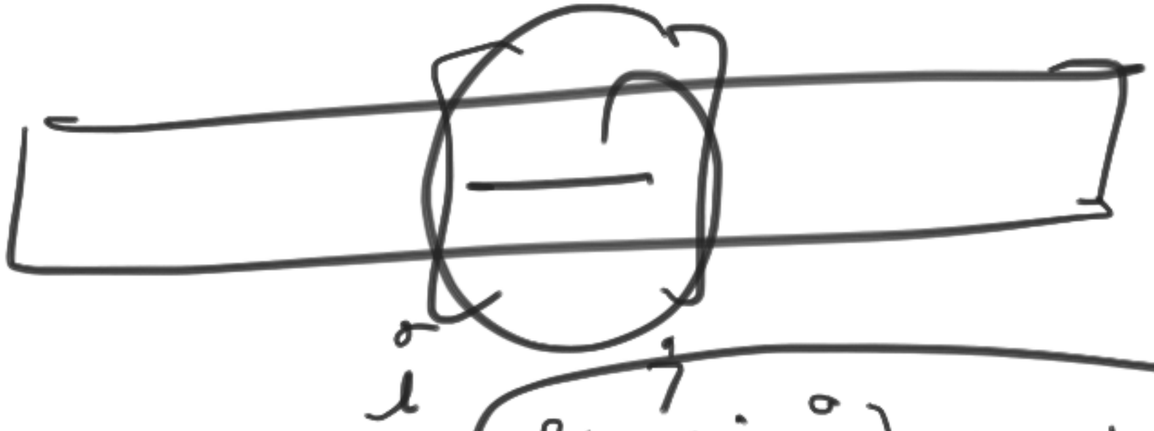
} return maxoneCount

```
for (r = 0 to n-1)
{   if (A[r] == 0)
        2C++
    if (2C > K)
    {   if (A[l] == 0)
            2C--
        l++
    }
    return r - l
}
```



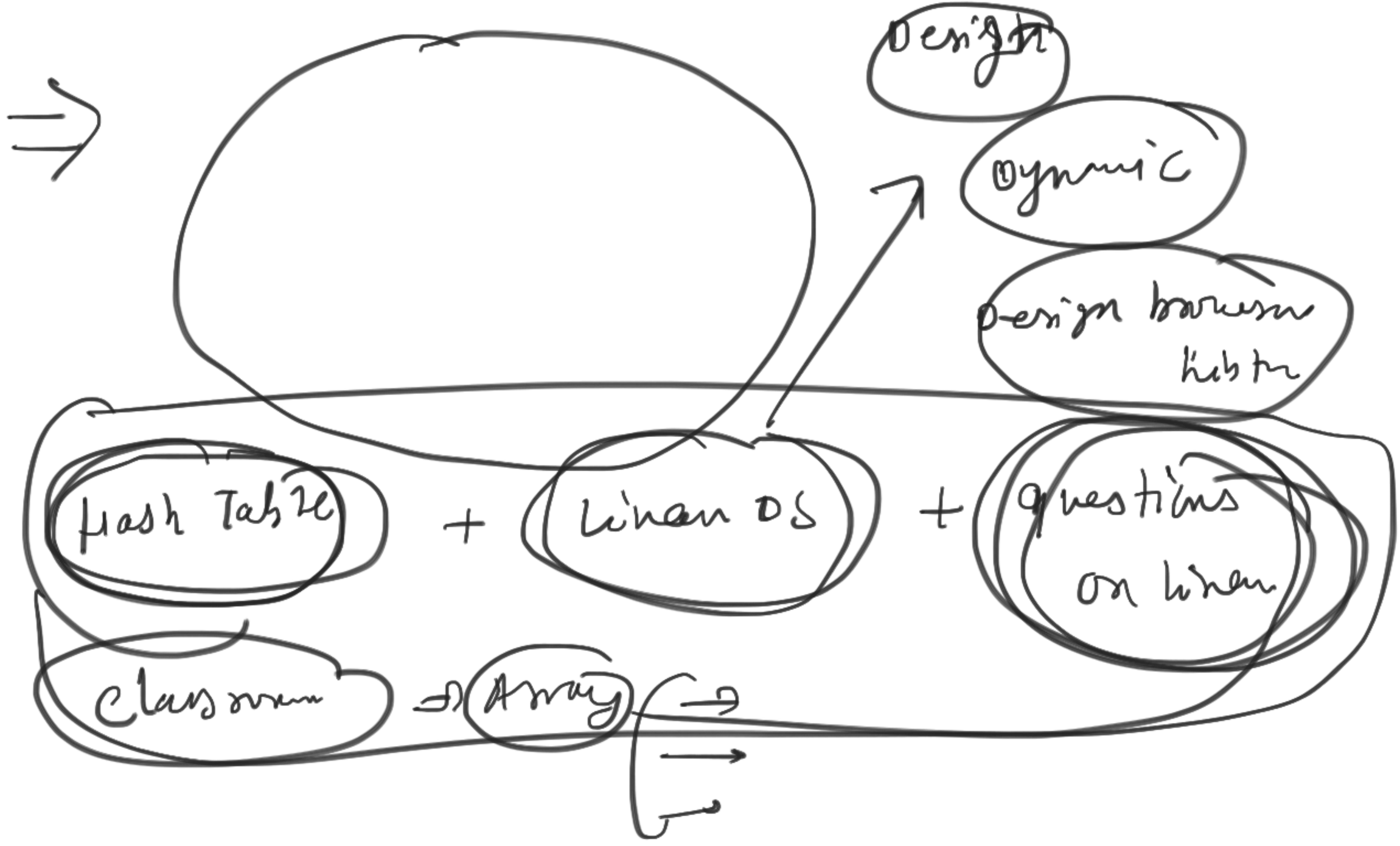
min size subarray sum

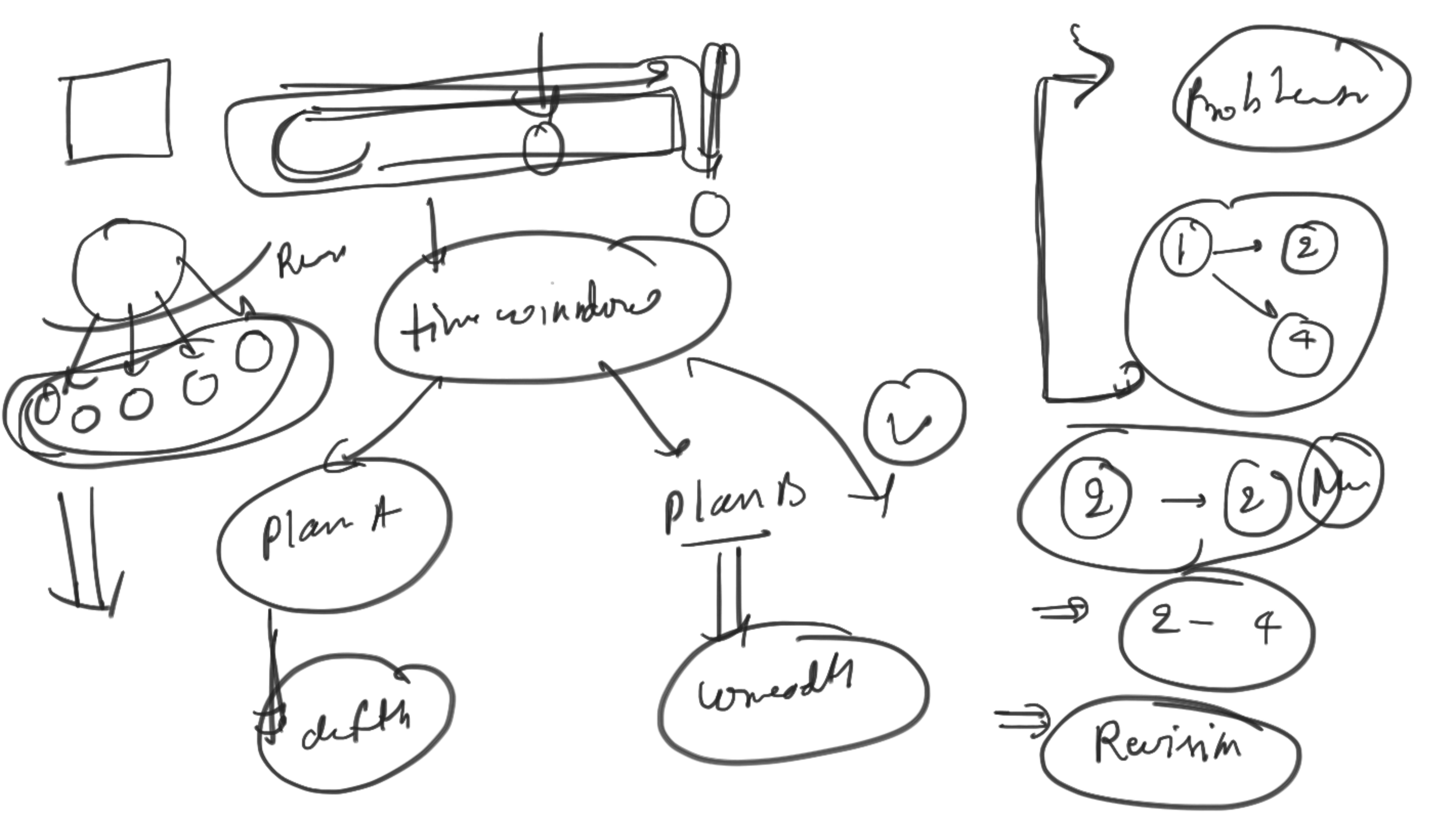
target



sum(i, j) ≥ target

min(j − i)


Design
Dynamic
Design browser history
Hash Table
+
Linear DS
+
questions on linear
Classroom
Array



Rev
time window
Plan A
Plan B
depth
Probleme
1
2
4
2
2
2 - 4
Revision