BONNE

Chandeleur

FEBRUARY 2

Noha warda

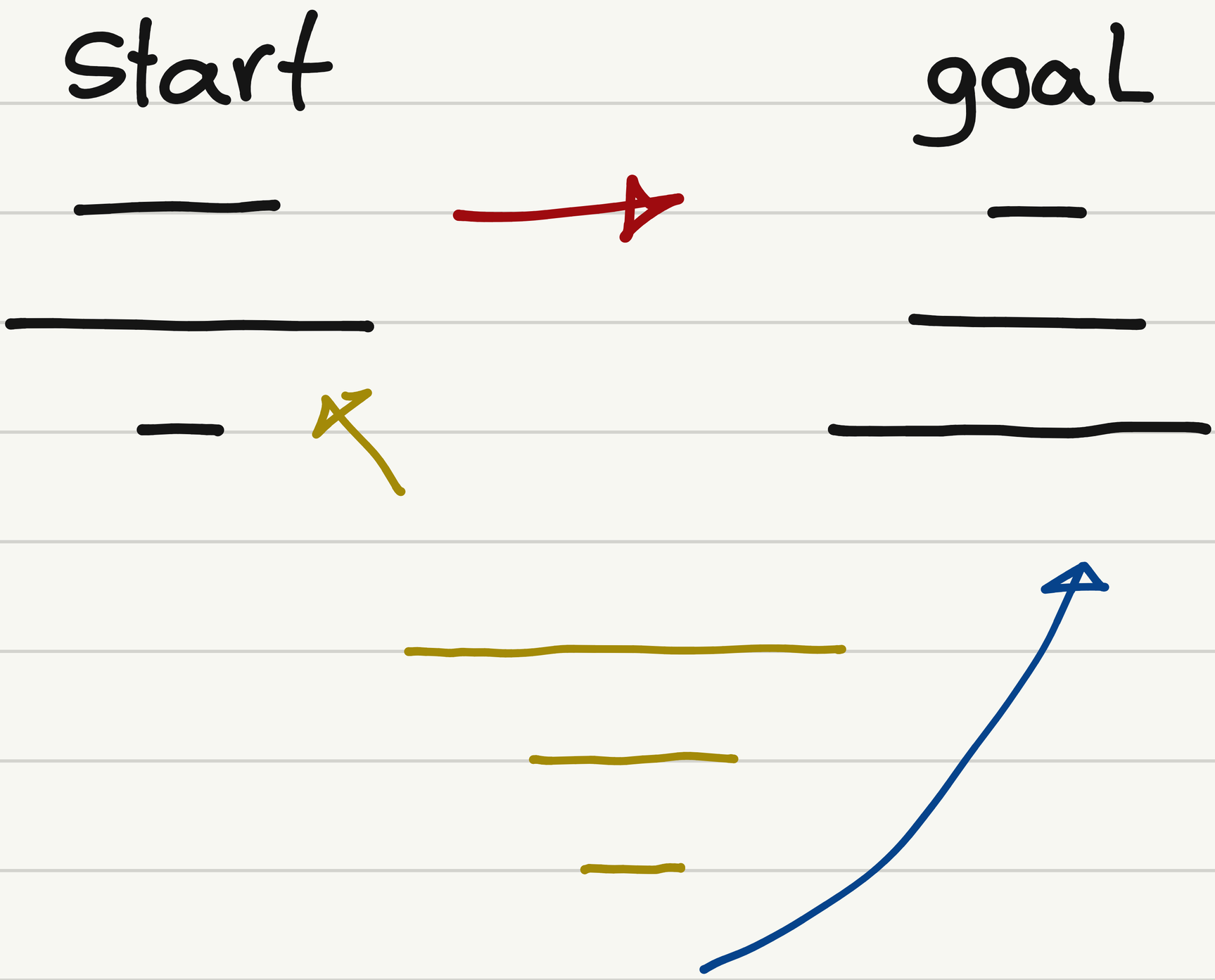
Flipping pancakes

Flipping pancakes
(decrease and conquer strategy)

* 2 examples
* code (python)
* space and time complexity

Start

goal



ex (2)

only one operation is allowed

Flip (o,i)

19 23 6 15 45 30 14

45 15 6 23 19 30 14

14 30 19 23 6 15 45

30 14 19 23 6 15 45 ← Fixed

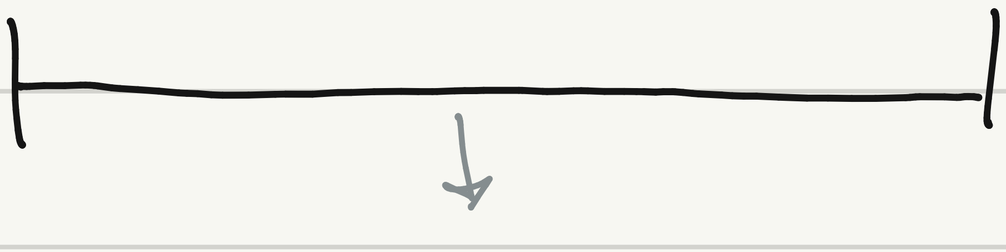
15 6 23 19 14 30 45 Fixed

23 6 15 19 14 30 45 Fixed

14 19 15 6

23 30 45 Fixed

19 14 15 6



23 30 45 Fixed

6 15 14

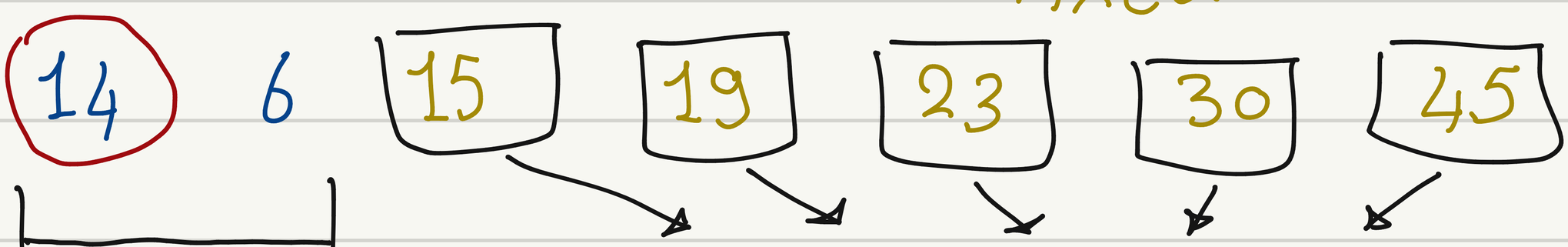
19 23 30 45

Fixed

15 6 14

Fixed

14 6 15 19 23 30 45



Fixed

6 14 15 19 23 30 45

```
def flip(arr, i):
    start = 0
    while start < i:
        temp = arr[start]
        arr[start] = arr[i]
        arr[i] = temp
        start += 1
        i -= 1
def findMax(arr, n):
    mi = 1
    for i in range(0,n+1):
        if arr[i] > arr[mi]:
            mi = i
    return mi
```

```
def pancakeSort(arr, n):
    curr_size = n
    while curr_size > 1:
        mi = findMax(arr, curr_size-1)
        if mi != curr_size-1:
            flip(arr, mi)
            flip(arr, curr_size-1)
        curr_size -= 1
def printArray(arr, n):
    for i in range(0,n):
        print ("%d"%( arr[i]),end=" ")
arr = [23, 10, 20, 11, 12, 6, 7]
n = len(arr)
pancakeSort(arr, n);
print ("Sorted Array ")
printArray(arr,n)
```

```
Sorted Array
6 7 10 11 12 20 23
```

* Space complexity:

  In-place sorting

  O(1)

* Time complexity:

  * Finding Maximum element O(n)

  * The Fliping Function that is called twice in each iteration of the main Loop, O(i) for each call

    O(n^2)

Thank

END

you

watching

for