hw2

110590002 王熯竑

1.

a

```
function sort(list):
  for i in 0 to length(list):
    if list[i] =='black':
      continue
    for j in range(0,i):
      if list[i-j-1]=='white':
        break
      else:
        swap(list[i-j],list[i-j-1])
```

the number of moves it makes is n(n+1)/2

b

if want to make the dark, light, dark, light, and so on, to all the light disks to the left-hand end

the n-th light disk must moved to the n-th position

if n-th light disk moved to a position which is less than n mean that :

1. a (< n)-th light disk is at the right-side of n-th light disk, that is, a swap happen between them, the swap is non-effective

if n-th light disk moved to position which is greater than n mean two state:

1. that the black disk is between 0 to n, that is illegal state
2. a (> n)-st light disk is between 0 to n, that is mean n move to n-1 position state happen.

because above, the n-th light disk must moved to n-th position

the light disk origin position is 2n

so the number of moves is $\sum(i=0)^{i=n}(2i-i)=\frac{1+n}{2}*n$

it is lower bound

2.

$$
\begin{aligned}
T(n) &= 2T\left(\frac{n}{2}\right)+\frac{n}{\lg(n)} \\
&= 2\left(2T\left(\frac{n}{4}\right)+\frac{\frac{n}{2}}{\lg(\frac{n}{2})}\right)+\frac{n}{\lg(n)} \\
&= 4T\left(\frac{n}{4}\right)+\frac{n}{\lg(n)-\lg(2)}+\frac{n}{\lg(n)} \\
&= 2^{\{\lg(n)\}}T\left(\frac{n}{2^{\lg(n)}}\right)+n\sum_{i=0}^{\lg(n)}\frac{1}{\lg(n)-i} \\
&= nT(1)+\frac{n}{2}\left(\frac{1}{\lg(n)}+\frac{1}{1}\right)\lg(n) \\
&= n+n\frac{1}{2}+n\frac{\lg(n)}{2}
\end{aligned}
$$

$T(n) = O(n \lg(n))$

mathematical induction

let $T(n) \leq n \lg(n)$ is true

then $T\left(\frac{n}{2}\right) \leq \frac{n}{2} \lg\left(\frac{n}{2}\right)$ is true

$$T(n) = 2T\left(\frac{n}{2}\right) + \frac{n}{\lg(n)} \leq n \lg\left(\frac{n}{2}\right) + \frac{n}{\lg(n)}$$

$$= n \lg(n) - n + \left(\frac{n}{\lg(n)}\right) \leq n \lg(n)$$

Q.E.D

3.

- $T(n) = 2T\left(\frac{n}{2}\right) + c$
  - $a = 2, b = 2, f(n) = c$
  - $n^{\lg_b a} = n$
  - $h(m) = \frac{c}{m} = cm^{-1}$
  - $-1 < 0$
  - $T(n) = O(n)$
- $T(n) = 4T\left(\frac{n}{2}\right) + cn$
  - $a = 4, b = 2, f(n) = c$
  - $n^{\lg_b a} = n^2$
  - $h(m) = \frac{cm}{m^2} = cm^{-1}$
  - $-1 < 0$
  - $T(n) = O(n^2)$
- $T(n) = 4T\left(\frac{n}{2}\right) + cn^3$
  - $a = 4, b = 2, f(n) = cn^3$
  - $n^{\lg_b a} = n^2$
  - $h(m) = \frac{cm^3}{m^2} = cm$
  - $1 > 0$
  - $T(n) = n^2 O(cn) = O(n^3)$
- $T(n) = 4T\left(\frac{n}{3}\right) + cn$
  - $a = 4, b = 3, f(n) = cn$
  - $n^{\lg_b a} = n^{\lg_3 4}$
  - $h(m) = \frac{cm}{m^{\lg_3 4}} = cm^{1 - \lg_3 4}$
  - $1 - \lg_3 4 < 0$
  - $T(n) = O\left(n^{\lg_3 4}\right)$

result
$O(n) < O\left(n^{\lg_3 4}\right) < O(n^2) < O(n^3)$

$\Rightarrow 2T\left(\frac{n}{2}\right) + c < 4T\left(\frac{n}{3}\right) + cn < 4T\left(\frac{n}{2}\right) + cn < 4T\left(\frac{n}{2}\right) + cn^3$

4.

- x_l is left half of x.
- x_r is right half of x.
- y_l is left half of y.
- y_r is right half of y.

- $x * y = x_l * 2^{\frac{n}{2}} * y_l * 2^{\frac{n}{2}} + x_l * 2^{\frac{n}{2}} * y_r + x_r * y_l * 2^{\frac{n}{2}} + x_r * y_r$
$$= x_l * y_l * (2^n) + (x_l * y_r + x_r * y_l) * 2^{\frac{n}{2}} + x_r * y_r$$
- $(x_l + x_r) * (y_l + y_r) = x_l * y_l + x_l * y_r + x_r * y_l + x_r * y_r$
- $x_l * y_r + x_r * y_l = (x_l + x_r) * (y_l + y_r) - x_l * y_l - x_r * y_r$
- $x * y = x_l * y_l * (2^n) + (x_l * y_r + x_r * y_l) * 2^{\frac{n}{2}} + x_r * y_r$
$$= x_l * y_l * (2^n) + ((x_l + x_r) * (y_l + y_r) - x_l * y_l - x_r * y_r) * 2^{\frac{n}{2}} + x_r * y_r$$
- $T(n) = F_1(x_l, y_l) + F_2(x_r, y_r) + F_3(x_l + x_r, y_l + y_r)$

pseudo-code

Mul(x,y): if x==0 and y==0: return 0 n = max(len(x),len(y))
if n == 1: return x*y $x_l \leftarrow$ left half of x.
$x_r \leftarrow$ right half of x.
$y_l \leftarrow$ left half of y.
$y_r \leftarrow$ right half of y.
r_1 = Mul(x_l, y_l)
r_2 = Mul(x_r, y_r)
r_3 = Mul($x_l + x_r, y_l + y_r$)
return $r_1 * 2^n + (r_3 - r_1 - r_2) * 2^{\frac{n}{2}} + r_2$

T(n)

$$\begin{aligned} T(n) &= 3T\left(\frac{n}{2}\right) + c \\ &a = 4, b = 3, f(n) = c \\ n^{\lg_b a} &= n^{\{\lg_3 4\}} \\ h(m) &= \frac{c}{m^{\{\lg_3 4\}}} = cm^{\{-\lg_3 4\}} \\ -\lg_3 4 &< 0 \\ T(n) &= O\left(n^{\lg_3 4}\right) \end{aligned}$$