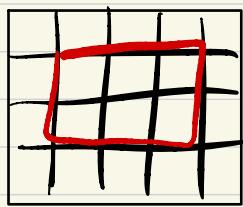
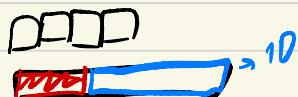


Algo  $\rightarrow Q \leq 2D$

Data Structure

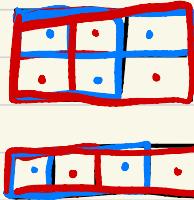


$O(n^2)$  per query

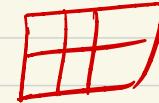
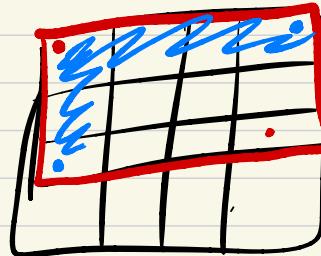
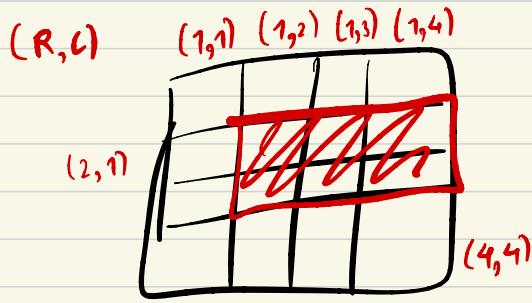
$O(1)$  per query

$O(n^2)$  precompute

$O(n^2 + Q)$ ;  $Q :=$  number of queries

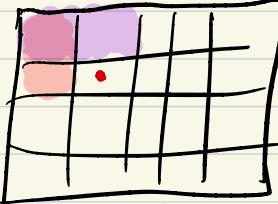


$$qs(1) - qs(2) - qs(4) + qs(3)$$



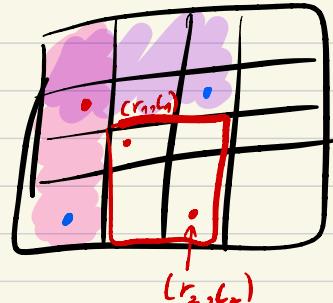
void precompute()

```
for (r; r > R) {  
    for (C; C > C) {  
        qs(r)(C) = qs(r-1)(C) + qs(r)(C-1) - qs(r-1)(C-1) + arr(r)(C)
```



```
int sum (int r1, int c1, int r2, int c2) {  
    if (r1 > r2 or c1 > c2) return -1  
    return qs(r2)(c2) - qs(r2)(c1-1) - qs(r1-1)(c2) + qs(r1-1)(c1-1)
```

?

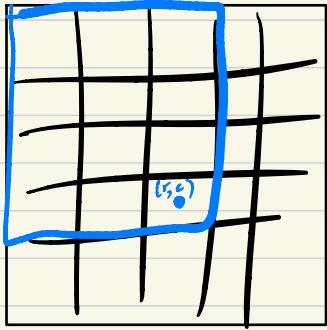


$(r_2, c_2)$

$(r_1-1, c_2)$

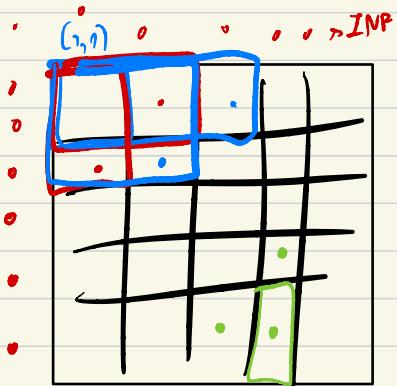
$(r_1-1, c_1-1)$

(i,j)



min from (i,j) to (r,c)

$O(R \times C)$  per query [Brute Force]



```

    mi(r,c) := minimum value from (i,j) to (r,c)
void precompute() {
    for (r: 0 → R) {
        for (c: 0 → C) {
            if (r == 0 or c == 0) mi(r,c) = INF
            else mi(r,c) = min { arr(r)(c), mi(r-1)(c), mi(r)(c-1) }
    }
}

```

```

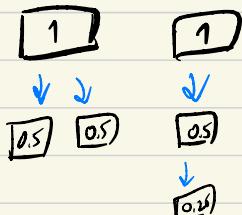
int query(int r, int c) {
    return mi(r,c)
}

```

min / max

& min / max inverse

## Divide and Conquer (D&C)



binary exponentiation  
Binary Exponentiation →

- Binary Search  $O(\log n)$



- Binary Exponentiation  $O(\log n)$

- Merge Sort  $O(n \log n)$

# Binary Exponentiation

$$a^n \text{ မျှ } 2^8 = \underbrace{2 \times 2 \times \dots \times 2}_{8 \text{ မျှ}} = O(n)$$

$$\begin{aligned} 2^8 &= 2^4 \times 2^4 & b = 2^4 \rightarrow b \times b \\ 2^4 &= 2^2 \times 2^2 \\ 2^2 &= 2^1 \times 2^1 & 8 \rightarrow 4 \rightarrow 2 \rightarrow 1 \end{aligned}$$

$$f(n) = a^n$$

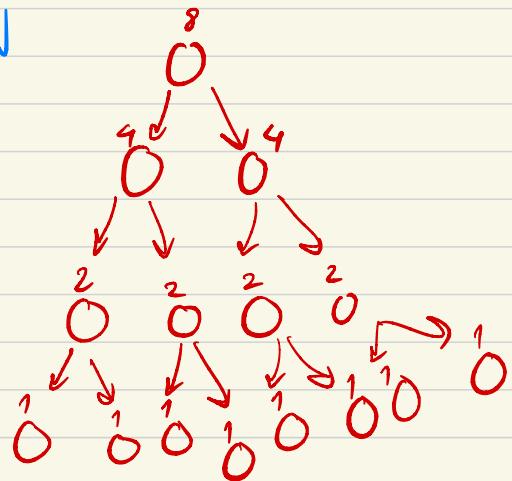
$$f(n) = \begin{cases} 1; n=0 \\ a; n=1 \\ f(n/2) \times f(n/2); n \text{ မျှ 2 } \\ f(n/2) \times f(n/2) \times a; n \text{ မျှ 4 } \end{cases}$$

```

define long long ll
ll expo(ll a, ll n){
    if (n==0) return 1
    if (n==1) return a
    ll b = expo(a, n/2)
    if (n%2 == 0){
        return expo(a, n/2) * expo(a, n/2)
    } else {
        return expo(a, n/2) * expo(a, n/2) * a
    }
}

```

$$\log_2 N$$



Pass by reference  
Merge sort linked

## Pass by Reference

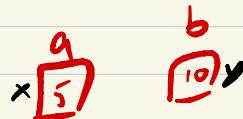
void swap( int &x , int &y){

int tmp = x

x = y

y = tmp

?

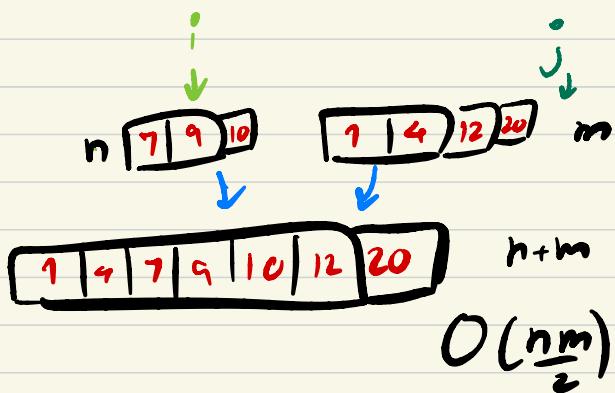
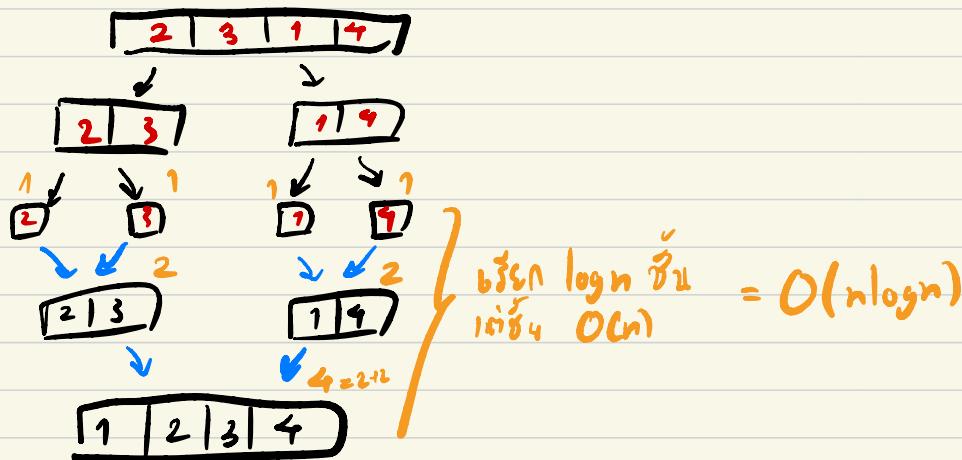


a=5, b=10

swap(a,b)

print (a,b) // 10,5

## Merge Sort $\rightarrow O(n \log n)$



```

int arr()
void merge_sort (int l, int r){ // l = leftmost index
    r = rightmost index
    if (l == r) return
    int mid = (l+r)/2
    merge_sort(l, mid), merge_sort(mid+1, r)

    int i=l, j=mid+1, tmp[l-(l+1)] = { }, k=0 // k is index of tmp
    while (i < mid and j < r){
        if (arr[i] <= arr[j]){
            tmp[k] = arr[i]
            i++
        }
        else {
            tmp[k] = arr[j]
            j++
        }
    }

    while (i < mid){
        tmp[k] = arr[i]
        k++, i++
    }

    while (j < r){
        tmp[k] = arr[j]
        k++, j++
    }

    for (k=0; k < r-l+1; k++){
        arr[m] = tmp[k]
    }
}

```

merge sort ( $0, n-1$ )

# Master Theorem

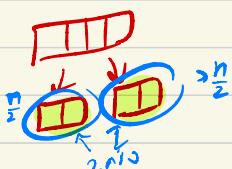
$$T(n) = \begin{cases} O(1); & n=1 \\ a T(\frac{n}{b}) + O(n^d); & n>1 \end{cases}$$

$$O(n^d); \quad a < b^d$$

$$O(n^d \log n); \quad a = b^d$$

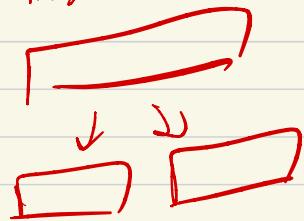
$$O(n^{\log_b a}); \quad a > b^d$$

*mergesort*  $\rightarrow O(1); n=1$



$$2T\left(\frac{n}{2}\right) + O(n); \quad n>1$$

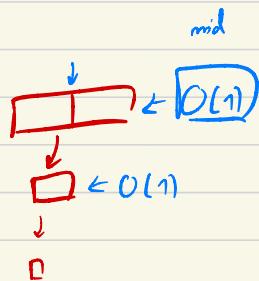
mid  $= \underline{l+r}$   $\xrightarrow{\text{mergesort}} \frac{n}{2}$   
 mergesort  $(l, \underline{mid})$  ①  $\frac{n}{2}$   $a=2$   
 mergesort  $(mid+1, r)$  ②  $\frac{n}{2}$   $b=2$



$$a=2, b=2, d=1 \rightarrow a=2, b^d=2 \rightarrow \text{case } ② \nrightarrow a=b^d$$

$$O(n \log n) = O(n \log n) \checkmark$$

*binary search*  $\rightarrow O(1); n=1$

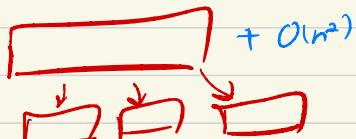


$$1T\left(\frac{n}{2}\right) + O(1); \quad n>1$$

$$a=1, b=2, d=0 \rightarrow a=1, b^d=1 \rightarrow \text{case } ② \nrightarrow a=b^d$$

$$O(n \log n) = O(n \log n)$$

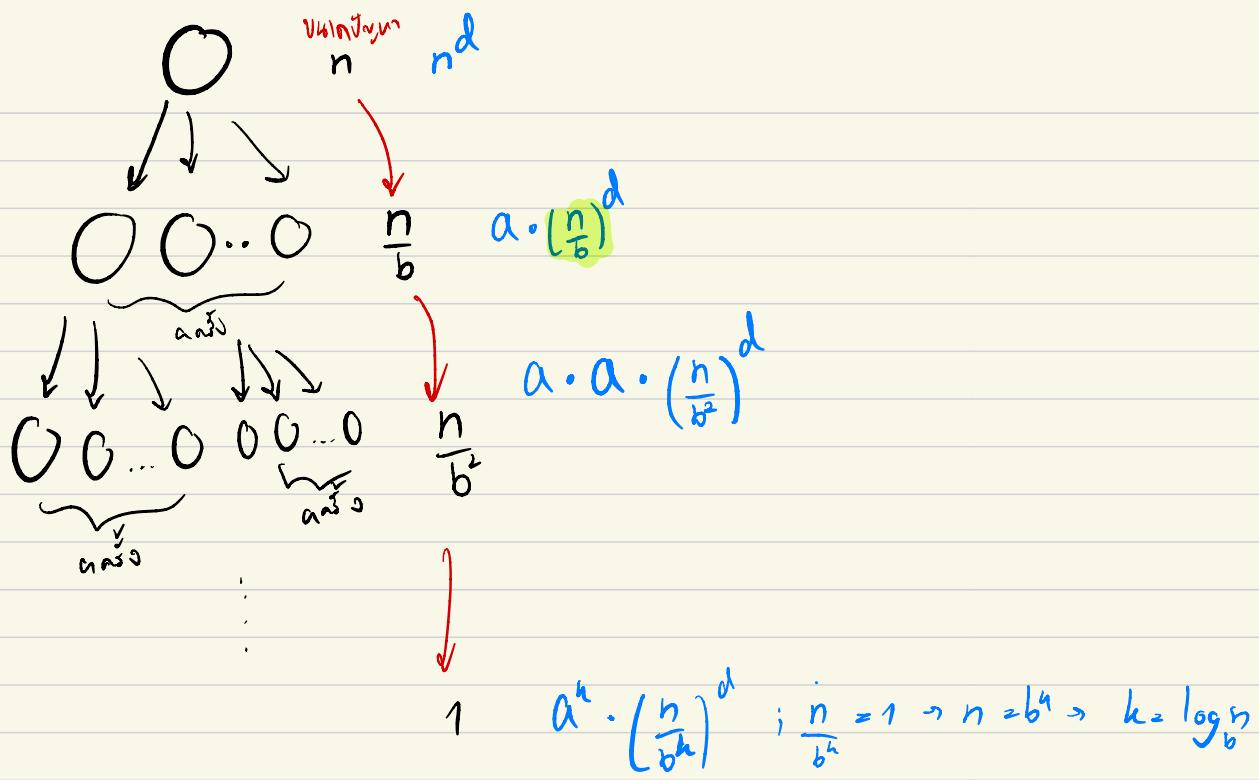
$T(n)$   $\rightarrow O(1); n=1$



$$3T\left(\frac{n}{2}\right) + O(n^2); \quad n>1$$

$$a=3, b=2, d=2 \rightarrow a=3, b^d=4 \nrightarrow \text{case } ① \quad a < b^d$$

$$O(n^d) = O(n^2)$$



$$= n^d + a \left(\frac{n}{b}\right)^d + a^2 \left(\frac{n}{b^2}\right)^d + \dots + a^k \left(\frac{n}{b^k}\right)^d$$

$$= n^d + a \frac{n^d}{b^d} + a^2 \frac{n^d}{b^{2d}} + \dots + a^k \frac{n^d}{b^{kd}}$$

$$= n^d \left[ 1 + \frac{a}{b^d} + \frac{a^2}{b^{2d}} + \dots + \frac{a^k}{b^{kd}} \right]$$

$$= n^d \cdot \sum_{h=0}^{\log_b n} \left(\frac{a}{b^d}\right)^h$$

Case(3)  $\frac{a}{b^d} < 1$

$$\sum_{h=0}^{\log_b n} \left(\frac{a}{b^d}\right)^h = O(n^d)$$

$\frac{a}{b^d} = 1$

$$\sum_{h=0}^{\log_b n} \left(\frac{a}{b^d}\right)^h = 1 + 1 + \dots + 1 = \log_b n \cdot 1 = O(n \log n)$$

Case(3)  $\frac{a}{b^d} > 1$

အောက်မှာ ပေါ်လောက်နိုင် ရှိ မြတ်စွာ 1, 3, 5, 7 သို့ မြတ်စွာ 2, 6, 10, 14 ရှိ မြတ်စွာ 2, 6, 18, 54, 162, 486, 1458, 4374, 13122, 39366, 118098, 354294, 1062882, 3188646, 9565938, 28697814, 86093442, 258280326, 774840978, 2324522934, 7073568802, 21220706406, 63662119218, 190986357654, 572959072962, 1718877218886, 5156631656658, 15469894969974, 46409684909922, 139229054729766, 417687164189298, 1253061492567994, 3769184477693982, 11307553433081946, 34922660299245838, 104768080897737514, 314304242693212542, 943012727079637626, 2829038181238912878, 8487114543716738624, 25461343631150215872, 76384030893450647616, 229152192678351942848, 707456577935055828544, 2122369733805167485328, 6367109201415492455984, 19101327604246477367944, 57303982812739432103888, 171911948438218296311664, 515735845314654898935008, 1547207535943964696785024, 4641622607831903890355072, 13924867823505711670855216, 41774593460517135012565648, 12532378038155336503766688, 37691734114366019511299664, 11307524337319805853389888, 34922642387959419660169632, 10476812716389805898050896, 31430414663699419294152688, 94301257512331417647175376, 28290379553799435284122616, 8487112266999811582437584, 25461329169999434748112752, 7638401522999931424433824, 2122369584999997243210976, 6367108914999974140132928, 1910132352999991240396944, 5730397574999973120119888, 1719119021999990640339776, 5157357942999971680109552, 1547207309999968320309888, 4641622571999964640119776, 1392486738999962320069552, 4177459156999961600139376, 1253237605999960320007872, 3769173073999959680002752, 1130752270999958880001536, 3492264041999957120001328, 1047681237999956000001120, 3143041311999954400000912, 9430125651999953200000704, 2829037783999952000000496, 8487112121999951200000384, 2546132554999950400000272, 7638401353999949600000264, 2122369326999948800000256, 6367108688999948000000248, 19101320589999472000000240, 57303973209999464000000232, 17191188929999456000000224, 51573570529999448000000216, 15472071249999440000000208, 46416220769999432000000200, 13924865989999424000000192, 41774590309999416000000184, 12532375729999408000000176, 37691730049999400000000168, 113075223699993920000000160, 349226403699993840000000152, 104768121899993760000000144, 314304130899993680000000136, 943012564099993600000000128, 282903777099993520000000120, 848711207299993440000000112, 254613253299993360000000104, 763840133499993280000000096, 212236930499993200000000088, 636710866699993120000000080, 191013207699993040000000072, 573039730899992960000000064, 171911884899992880000000056, 515735704099992800000000048, 154720711099992720000000040, 464162206299992640000000032, 139248647299992560000000024, 417745900499992480000000016, 125323753499992400000000008, 376917300699992320000000000, 113075223699992240000000000, 349226403699992160000000000, 104768121899992080000000000, 314304130899992000000000000, 943012564099991920000000000, 282903777099991840000000000, 848711207299991760000000000, 254613253299991680000000000, 763840133499991600000000000, 212236930499991520000000000, 636710866699991440000000000, 191013207699991360000000000, 573039730899991280000000000, 171911884899991200000000000, 515735704099991120000000000, 154720711099991040000000000, 464162206299990960000000000, 139248647299990880000000000, 417745900499990800000000000, 125323753499990720000000000, 376917300699990640000000000, 113075223699990560000000000, 349226403699990480000000000, 104768121899990400000000000, 314304130899990320000000000, 943012564099990240000000000, 282903777099990160000000000, 848711207299990080000000000, 254613253299990000000000000, 763840133499999920000000000, 212236930499999840000000000, 636710866699999760000000000, 191013207699999680000000000, 573039730899999600000000000, 171911884899999520000000000, 515735704099999440000000000, 154720711099999360000000000, 464162206299999280000000000, 139248647299999200000000000, 417745900499999120000000000, 125323753499999040000000000, 376917300699999000000000000, 113075223699998920000000000, 349226403699998840000000000, 104768121899998760000000000, 314304130899998680000000000, 943012564099998600000000000, 282903777099998520000000000, 848711207299998440000000000, 254613253299998360000000000, 763840133499998280000000000, 212236930499998200000000000, 636710866699998120000000000, 191013207699998040000000000, 573039730899997960000000000, 171911884899997880000000000, 515735704099997800000000000, 154720711099997720000000000, 464162206299997640000000000, 139248647299997560000000000, 417745900499997480000000000, 125323753499997400000000000, 376917300699997320000000000, 113075223699997240000000000, 349226403699997160000000000, 104768121899997080000000000, 314304130899997000000000000, 943012564099996920000000000, 282903777099996840000000000, 848711207299996760000000000, 254613253299996680000000000, 763840133499996600000000000, 212236930499996520000000000, 636710866699996440000000000, 191013207699996360000000000, 573039730899996280000000000, 171911884899996200000000000, 515735704099996120000000000, 154720711099996040000000000, 464162206299995960000000000, 139248647299995880000000000, 417745900499995800000000000, 125323753499995720000000000, 376917300699995640000000000, 113075223699995560000000000, 349226403699995480000000000, 104768121899995400000000000, 314304130899995320000000000, 943012564099995240000000000, 282903777099995160000000000, 848711207299995080000000000, 254613253299995000000000000, 763840133499994920000000000, 212236930499994840000000000, 636710866699994760000000000, 191013207699994680000000000, 573039730899994600000000000, 171911884899994520000000000, 515735704099994440000000000, 154720711099994360000000000, 464162206299994280000000000, 139248647299994200000000000, 417745900499994120000000000, 125323753499994040000000000, 376917300699993960000000000, 113075223699993880000000000, 349226403699993800000000000, 104768121899993720000000000, 314304130899993640000000000, 943012564099993560000000000, 282903777099993480000000000, 848711207299993400000000000, 254613253299993320000000000, 763840133499993240000000000, 212236930499993160000000000, 636710866699993080000000000, 191013207699993000000000000, 573039730899992920000000000, 171911884899992840000000000, 515735704099992760000000000, 154720711099992680000000000, 464162206299992600000000000, 139248647299992520000000000, 417745900499992440000000000, 125323753499992360000000000, 376917300699992280000000000, 113075223699992200000000000, 349226403699992120000000000, 104768121899992040000000000, 314304130899991960000000000, 943012564099991880000000000, 282903777099991800000000000, 848711207299991720000000000, 254613253299991640000000000, 763840133499991560000000000, 212236930499991480000000000, 636710866699991400000000000, 191013207699991320000000000, 573039730899991240000000000, 171911884899991160000000000, 515735704099991080000000000, 154720711099991000000000000, 464162206299990920000000000, 139248647299990840000000000, 417745900499990760000000000, 125323753499990680000000000, 376917300699990600000000000, 113075223699990520000000000, 349226403699990440000000000, 104768121899990360000000000, 314304130899990280000000000, 943012564099990200000000000, 282903777099990120000000000, 848711207299990040000000000, 254613253299990000000000000, 763840133499999920000000000, 212236930499999840000000000, 636710866699999760000000000, 191013207699999680000000000, 573039730899999600000000000, 171911884899999520000000000, 515735704099999440000000000, 154720711099999360000000000, 464162206299999280000000000, 139248647299999200000000000, 417745900499999120000000000, 125323753499999040000000000, 376917300699998960000000000, 113075223699998880000000000, 349226403699998800000000000, 104768121899998720000000000, 314304130899998640000000000, 943012564099998560000000000, 282903777099998480000000000, 848711207299998400000000000, 254613253299998320000000000, 763840133499998240000000000, 212236930499998160000000000, 636710866699998080000000000, 191013207699998000000000000, 573039730899997920000000000, 171911884899997840000000000, 515735704099997760000000000, 154720711099997680000000000, 464162206299997600000000000, 139248647299997520000000000, 417745900499997440000000000, 125323753499997360000000000, 376917300699997280000000000, 113075223699997200000000000, 349226403699997120000000000, 104768121899997040000000000, 314304130899996960000000000, 943012564099996880000000000, 282903777099996800000000000, 848711207299996720000000000, 254613253299996640000000000, 763840133499996560000000000, 212236930499996480000000000, 636710866699996400000000000, 191013207699996320000000000, 573039730899996240000000000, 171911884899996160000000000, 515735704099996080000000000, 154720711099996000000000000, 464162206299995920000000000, 139248647299995840000000000, 417745900499995760000000000, 125323753499995680000000000, 376917300699995600000000000, 113075223699995520000000000, 349226403699995440000000000, 104768121899995360000000000, 314304130899995280000000000, 943012564099995200000000000, 282903777099995120000000000, 848711207299995040000000000, 254613253299994960000000000, 763840133499994880000000000, 212236930499994800000000000, 636710866699994720000000000, 191013207699994640000000000, 573039730899994560000000000, 171911884899994480000000000, 515735704099994400000000000, 154720711099994320000000000, 464162206299994240000000000, 139248647299994160000000000, 417745900499994080000000000, 125323753499994000000000000, 376917300699993920000000000, 113075223699993840000000000, 349226403699993760000000000, 104768121899993680000000000, 314304130899993600000000000, 943012564099993520000000000, 282903777099993440000000000, 848711207299993360000000000, 254613253299993280000000000, 763840133499993200000000000, 212236930499993120000000000, 636710866699993040000000000, 191013207699992960000000000, 573039730899992880000000000, 171911884899992800000000000, 515735704099992720000000000, 154720711099992640000000000, 464162206299992560000000000, 139248647299992480000000000, 417745900499992400000000000, 125323753499992320000000000, 376917300699992240000000000, 113075223699992160000000000, 349226403699992080000000000, 104768121899992000000000000, 314304130899991920000000000, 943012564099991840000000000, 282903777099991760000000000, 848711207299991680000000000, 254613253299991600000000000, 7638401334999915200000

### Casc(3)

$\frac{a}{b^d} > 1$  ເພື່ອຈຳເປັນການກວດສອບທີ່  $a_1 = 1$ ,  $r = \frac{a}{b^d}$

$$\text{ຖືນ } S_n = \frac{a_1(r^n - 1)}{r-1}; n := \text{ຈຸດທີ}$$

$$S_n = \frac{1 \left[ \left( \frac{a}{b^d} \right)^{\log_b^{n+1}} - 1 \right]}{\left( \frac{a}{b^d} \right) - 1}$$

$$S_n = \frac{\frac{a^{\log_b^{n+1}}}{b^{d\log_b^{n+1}}} - 1}{\frac{a}{b^d} - 1}$$

$$S_n = \frac{a^{\log_b^{n+1}} \circ b^d}{b^{d\log_b^{n+1}} \circ a}$$

$$S_n = \frac{a^{\log_b^n}}{b^{\log_b^d}}$$

$$\text{ຖືນ } y^{\log_x} = x; S_n = \frac{a^{\log_b^n}}{n^d}$$

$$\text{ຖືນ } n^d \cdot \sum_{k=0}^{\log_b^n} \left( \frac{a}{b^d} \right)^k \rightarrow \text{ຕະຫຼາດ } n^d \cdot \frac{a^{\log_b^n}}{a^{\log_b^n} - 1} = a^{\log_b^n}$$

$$\text{ຖືນ } y^{\log_x} = x^{\log_b y}; n^{\log_b a}$$

$$\text{ສິ່ງທີ່ } \frac{a}{b^d} > 1 \text{ ມີຄວາມ } O(n^{\log_b a})$$