Discrete Binary Search :)

Lecture 11 Binary Search

Lab about to 2:42 :

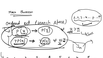
f(x) ↑↑
f(x) ↓↓
log(n)

K:

(n)
find → (log n)
and → 2 (log)

Map Reduce

Output pair (count, doc)



K( ) → value varies in 2.K or 2.K+1

P( ) → false

P(x) → true

P(x) → P(x+1) →

Q4

Partition → optimal answer

1) Better considered Q → feasible

2) Find maximum distance from one vertex to another

3) Find minimum weighted path from one vertex to all other vertices

Jump Search

O(Queries x K)

check(x) → check(5) ≥ x

check(x) → check(5) ≤ x

P: Find min time such that we can produce f product

B.C: Given a time x, can I produce f product in x

check(x) → yes → return x;

check(x) → no → check(x+1)

check(mid) → Time

O(n < range of ans)

O(n)

high = 10^{18}, low = 0, mid = (low+high)/2

while (high >= low){

if (check(mid)) {

ans = mid;

high = mid-1;

}

else {low = mid+1;}

mid = (high + low)/2;

}