

# Búsquedas

Reuso de material del Training Camp ICPC 2012

I Semestre - 2014

## ¿ Para qué sirve?

- Dada una lista ordenada, encontrar un número  $r$ .
- Búsqueda Lineal. Recorremos elemento por elemento y chequeamos si  $r$  está en la lista. Complejidad :  $O(n)$ .
- Búsqueda Binaria (BS). Tomamos el intervalo  $[a \dots b]$  nos fijamos si el elemento  $[(a + b)/2]$  es menor igual a  $r$  . Si lo es, solo me tengo que fijar si  $r$  está en  $[a, (a + b)/2]$ , sino  $r$  tiene que estar en  $[(a + b)/2, b]$ . Aplico recursivamente la misma idea. Complejidad :  $O(\log n)$

## Algoritmo

```
int BSearch(constvector < int > &arr, int r){  
    int left = 0, right = n;  
    while (right - left > 1){  
        int mid = (left + right)/2;  
        if (arr[mid] <= r){  
            left = mid;  
        } else {  
            right = mid;  
        }  
    }  
    return left};  
  
bool IsFound(constvector < int > &arr, int r){  
    return arr[BSearch(arr, r)] == r};
```

## Generalización de la Búsqueda Binaria

- Dado un intervalo  $[a, b]$ , y una propiedad  $P$  sobre el intervalo tal que  $\forall x, y \in [a, b]$ , si  $x < y$  entonces  $P(y) \rightarrow P(x)$ . BS nos encuentra el elemento más grande que cumple  $P$  o el elemento más chico que no cumple  $P$  (el siguiente del más grande que cumple  $P$ ).
- Demostración informal utilizando inducción. Supongamos que tenemos un intervalo  $[a, b]$  tal que cumple las condiciones dadas arriba. Además supongamos que hay al menos un elemento en  $[a, b]$  que cumple  $P$ .

Si  $[a, b]$  tiene un elemento, entonces ya tengo el elemento más grande que cumple  $P$ .

Si  $[a, b]$  tiene tamaño  $n > 1$ . Sea  $mid = (a + b)/2$ , si  $P(mid)$  vale, entonces como  $P$  es monótona estoy seguro que el elemento más grande que vale  $P$  está en  $[mid, b]$ , de lo contrario, estoy seguro de que esta en  $[a, mid]$ .

**Cont.**

- Dado  $n$  cajas tal que en cada caja entran  $c[i]$  elementos. Dados  $r$  elementos, distribuir los  $r$  elementos tal que la cantidad de elementos en la caja que más tiene sea mínimo. Devolver este valor.  
 $P(x)$  = puedo poner los  $r$  elementos de tal modo que cada caja no tiene más de  $x$  elementos.

## Cont.

```
assert( $r > 0 \&\& \sum_i(c[i]) \geq r$ );
int left = 0, right = r;
while(left < right - 1) {
    mid = (left + right)/2;
    if( P(mid, r) ) right = mid;
    else left = mid;
}
return right;

bool P(int x, int r) {
    int cnt = 0;
    for(int i = 0; i < n; i++) cnt += min(x, c[i]);
    return cnt >= r;}
```

Complejidad del algoritmo :  $O(n \log(n))$ , ya que en cada ciclo de la BS hago  $n$  pasos.

### Concepto: Búsqueda Ternaria

- Es una técnica para encontrar el máximo o mínimo en un intervalo de una función que es estrictamente creciente en un intervalo y luego estrictamente decreciente o viceversa.
- Se puede aplicar en funciones continuas o discretas.
- En funciones discretas es importante verificar que la función cumpla que sea estrictamente creciente y luego estrictamente decreciente. (no dos valores consecutivos iguales).

## Generalización

Asumamos que estamos buscando el máximo de una función  $f$ , y sabemos que el máximo está en el intervalo  $[A, B]$ . Para poder aplicar *TS*, debe existir algún valor  $x$  tal que

- Para todo  $a, b$  tal que  $A \leq a < b \leq x$ , tenemos  $f(a) < f(b)$
- Para todo  $a, b$  tal que  $x \leq a < b \leq B$ , tenemos  $f(a) > f(b)$

## Algoritmo

```
double TS(double A, double B) {  
    double left = A, right = B;  
    while(abs(right - left) < EPS) {  
        double lt = (2. * left + right)/3;  
        double rt = (left + 2. * right)/3;  
        if(f(lt) < f(rt)) left = lt;  
        else right = rt;}  
    return (left + right)/2;  
}
```

### Ejemplo

Supongamos que granjero John viven en una casa en la posición  $(x_h, y_h)$  con  $y_h > 0$ , y quiere regar su árbol que está en la posición  $(x_a, y_a)$ , con  $y_a > 0$ . Para poder regarlo, John camina hasta el río que se encuentra en el eje  $x$ , toma agua y se dirige hasta el árbol. ¿Cuál es la mínima distancia que tiene que recorrer granjero John para regar su árbol?