

Melis'in Listesi Çözümü

Sorudan kısaca bahsetmek gerekirse bize N uzunluğunda bir tamsayı dizisi ve K sayısı veriliyor.

Daha sonra bu dizide en uzun aralığı seçmemiz gerekiyor öyle ki, o aralıktaki tüm ikililerin farkı K sayısından küçük eşit olmalı.

Yapacağımız ilk çıkarım, seçtiğimiz aralıktaki tüm ikililere bakmak yerine o aralıktaki en büyük ve en küçük elemanın farkına bakmak yeterli olacaktır. Çünkü olabilecek en büyük fark max-min olacaktır.

Bir diğer çıkarım ise dizilerin max ve min farkını azaltabilmek için küçükten büyüğe sıralayıp gezmek olabilir.

Çözüm için ilk akla gelen yol, iç içe iki adet döngü kullanarak dizideki bütün mümkün aralıkları belirlemek daha sonra bu aralıktaki en büyük ve en küçük elemanı bulup farkını kontrol etmek olabilir. Bu çözüm N^3 'de çalışacağı için 12 puan alacaktır.

Her aralık için tekrar min ve max eleman bulmak yerine iç içe 2. Döngüde aralığı genişletirken max ve min elemanlarını güncellersek N^2 'ye düşürebiliriz bu da ekstra 14 puan demek.

Ancak asıl çözüm **binary search** veya **two pointer** konseptleri kullanarak çıkacaktır. Bu çözümde two pointer çözümünü açıklayacağım.

Aslında bu konseptte iç içe 2 döngü ile yaptığımız işlemi tek döngüye düşürüyoruz. Tek bir while döngüsü kullanarak 2 adet index tutuyoruz. Bu indexleri aralığımızın başı ve sonu olarak düşünebilirsiniz. O aralıktaki max ve min'i her eleman eklediğimde ve çıkardığımda güncelliyorum.

Güncelleme sonunda max-min değerim K'dan küçük eşitse anlıyorum ki ben bu aralığa daha eleman ekleme hakkım var. Bu durumda son indeximi 1 arttırıyorum ve bir sonraki elemanı da aralığımıza katmaya çalışıyorum.

Aksi takdirde max-min K'dan büyükse fazlalık var demektir bu yüzden baş indeximi 1 arttırıp en baştaki elemanı çıkartmış oluyorum.

Her güncellemede cevabımı maxlayarak sonuca ulaşmış oluyorum.

Çözüm Kodu:

Dil: C++

```
int main(){
    cin >> n >> k;
    for(int i=1;i<=n;i++)
        cin >> a[i];

    sort(a+1,a+n+1);
    int sz = 1;
    int mx = 1;
    int i=2;
    while( i <= n){
        if(a[i] - a[i-sz] <= k){
            i++;
            sz++;
            mx = max(mx,sz);
        }else{
            sz--;
            mx = max(mx,sz);
        }
    }
    cout << mx << endl;
}
```

Hasan Bal