

TURNUVA ÇÖZÜM

K sayısının bütün bölenlerini K iterasyonda bulabiliriz. \sqrt{K} 'ye kadar ilerlediğimiz bir for'da eğer $K\%i = 0$ ise i ve (K/i) birer bölendir. Her bölen için N uzunluğundaki dizimizi geziyoruz. Dizi elemanı a ise dizide $(b-a)$ 'dan kaç tane olduğuna bakıyoruz ($b = \text{bölen}$). Bunun için her sayıdan kaç tane olduğunu tuttuğumuz bir array kullanabiliriz. Her a için bunu yaptığımızda bu sayı için kaç farklı ihtimal olduğunu hesaplamış olduk. En son bütün diziyi gezip ihtimali en yüksek olan sayıyı ekrana bastırabiliriz. Zaman karmaşıklığı: $O(N \cdot \sqrt{K})$

Çözüm Kodu

```
#include <bits/stdc++.h>

using namespace std;

int main() {

    int t; cin >> t;
    while (t--) {

        int n, k;
        cin >> n >> k;
        vector<int> arr(n);
        vector<int> count(k, 0);

        int mininput = 1000000001;

        for (int i = 0; i < n; i++) {
            cin >> arr[i];
            mininput = min(mininput, arr[i]);
            if (arr[i] < k) count[arr[i]]++;
        }

        vector<int> ans(k+1, 0);
        for (int i = 1; i*i <= k; i++) {
            if (k%i == 0) {
                int last = -1;
                for (int a: arr) {
                    if (a == last) continue;
                    last = a;
                    if (i > a)
                        ans[a] += count[i-a];
                    if (i*i != k) {
                        if (k/i > a)
                            ans[a] += count[k/i-a];
                    }
                }
            }
        }

        int maxx = -1;
        int maxv = mininput;
```

```
        for (int i = 0; i < k; i++) {  
            if (count[i] && maxx < ans[i]) {  
                maxx = ans[i];  
                maxv = i;  
            }  
        }  
        cout << maxv << endl;  
    }  
}
```