**Algoritma Analizi Lab Uygulaması**

Emre Turgut - 170418024

1. **Bir Dizinin Modunu bulma**

Her iki algoritmada verilen diziden dizinin içerisindeki elemanların sıklıklarına göre maxHeap oluşturup daha sonrasında bu maxHeap’in en yukarısındaki elemanı yani en çok tekrar eden elemanı geri dönüyor. İki algoritma arasındaki tek fark ilkinin dizisi karışık iken ikincisinin sıralı olması.

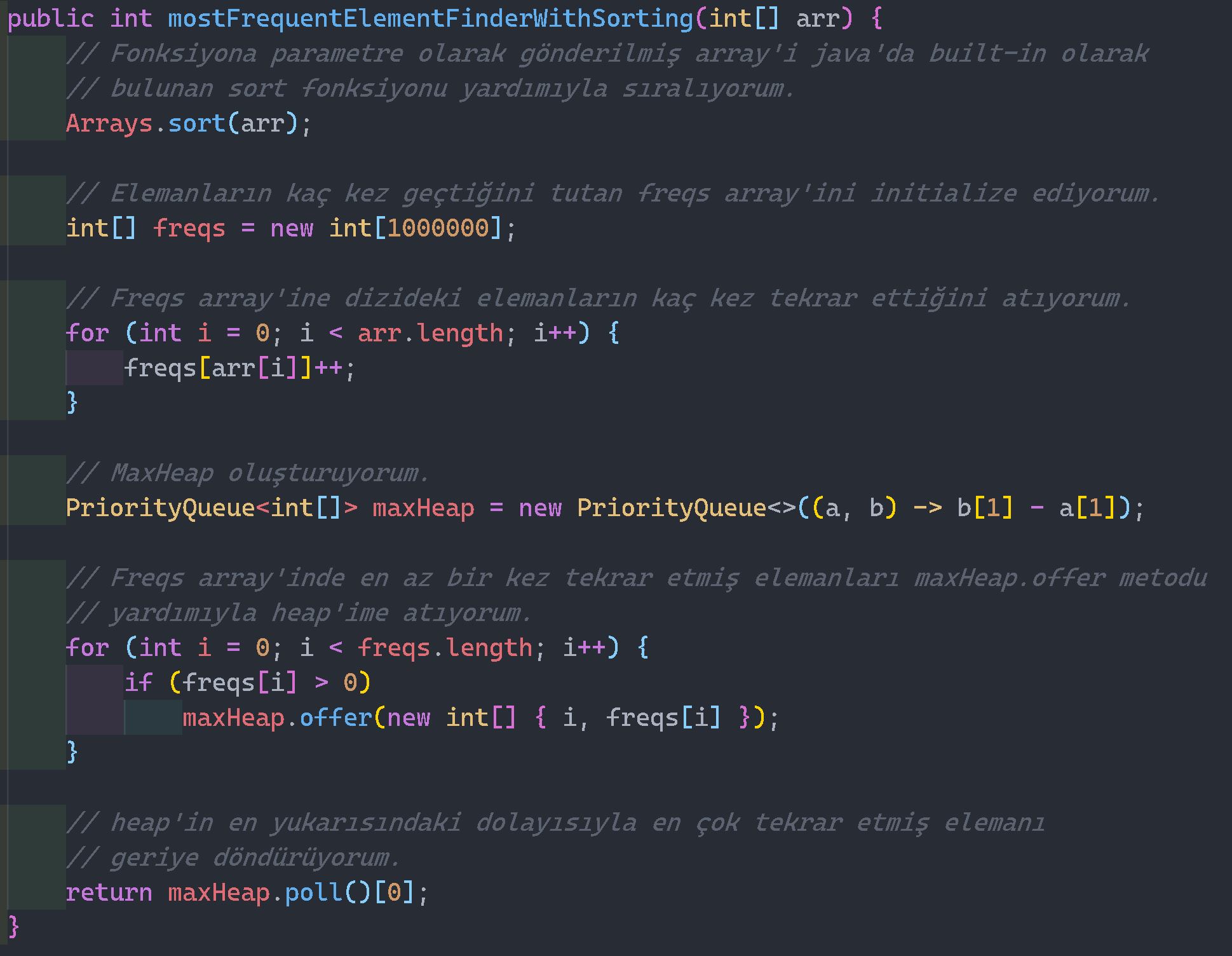


**Şekil 1 - Sıralanmamış dizide en çok tekrar edilen elemanı bulma.**

**Cop(n)** = En çok tekrar edilen işlem heap’in içerisine bir değer yüklemek.

***Cn(n) = nlogn***

***O(nlogn)***



**Şekil 2 - Sıralanmış dizide en çok tekrar edilen elemanı bulma.**

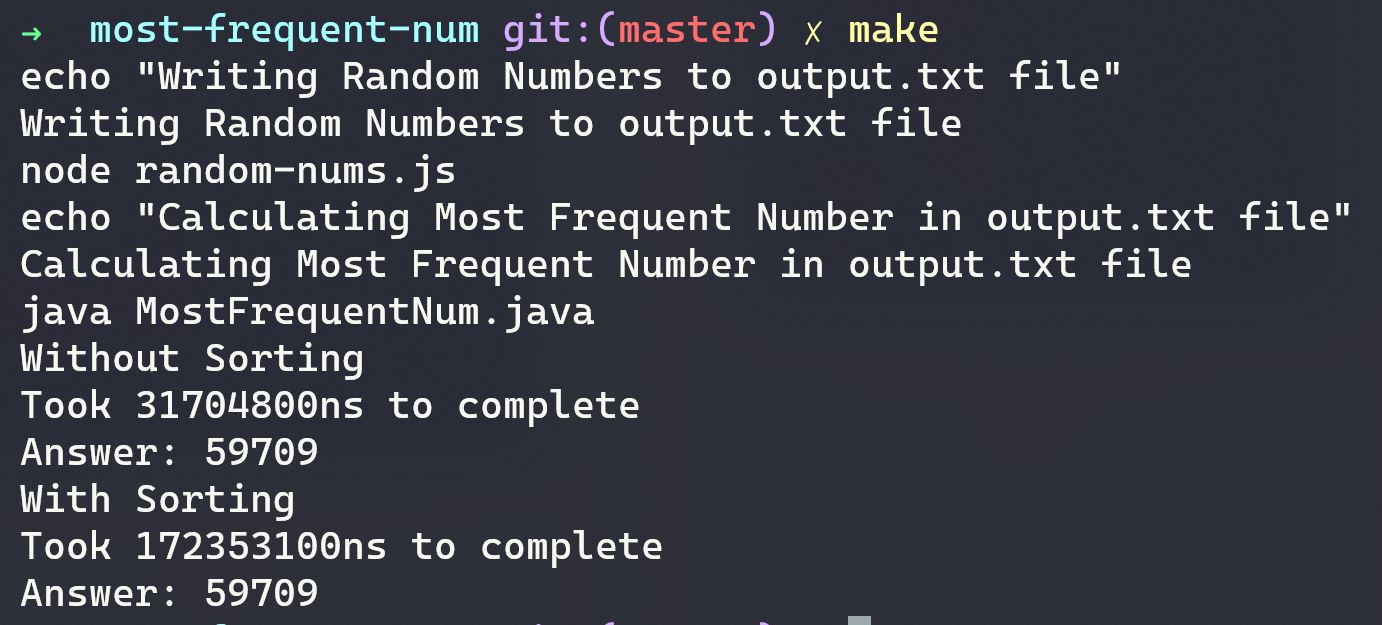
**Cop(n)** = En çok tekrar edilen işlem heap’in içerisine bir değer yüklemek.

***Cn(n) = nlogn***

***O(nlogn)***



**Şekil 3 - Dosyadan okuma işlemleri ve algoritma çalışma zamanıyla ilgili hesaplamaların yapıldığı main fonksiyonu.**

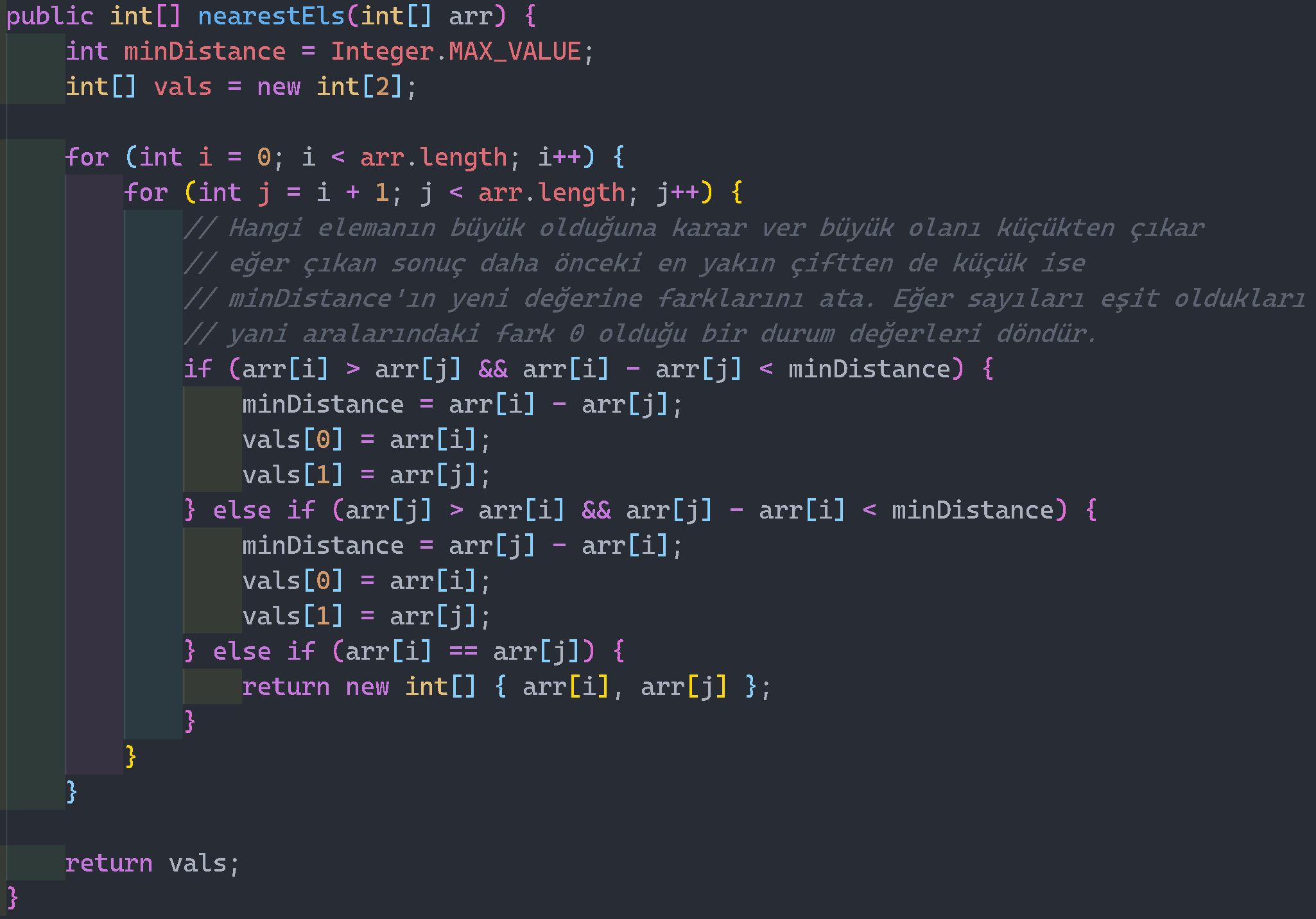


**Şekil 4 - Dizideki en çok tekrar eden elemanı bulan algoritmaların çalıştırılması ve sonuçlar.**

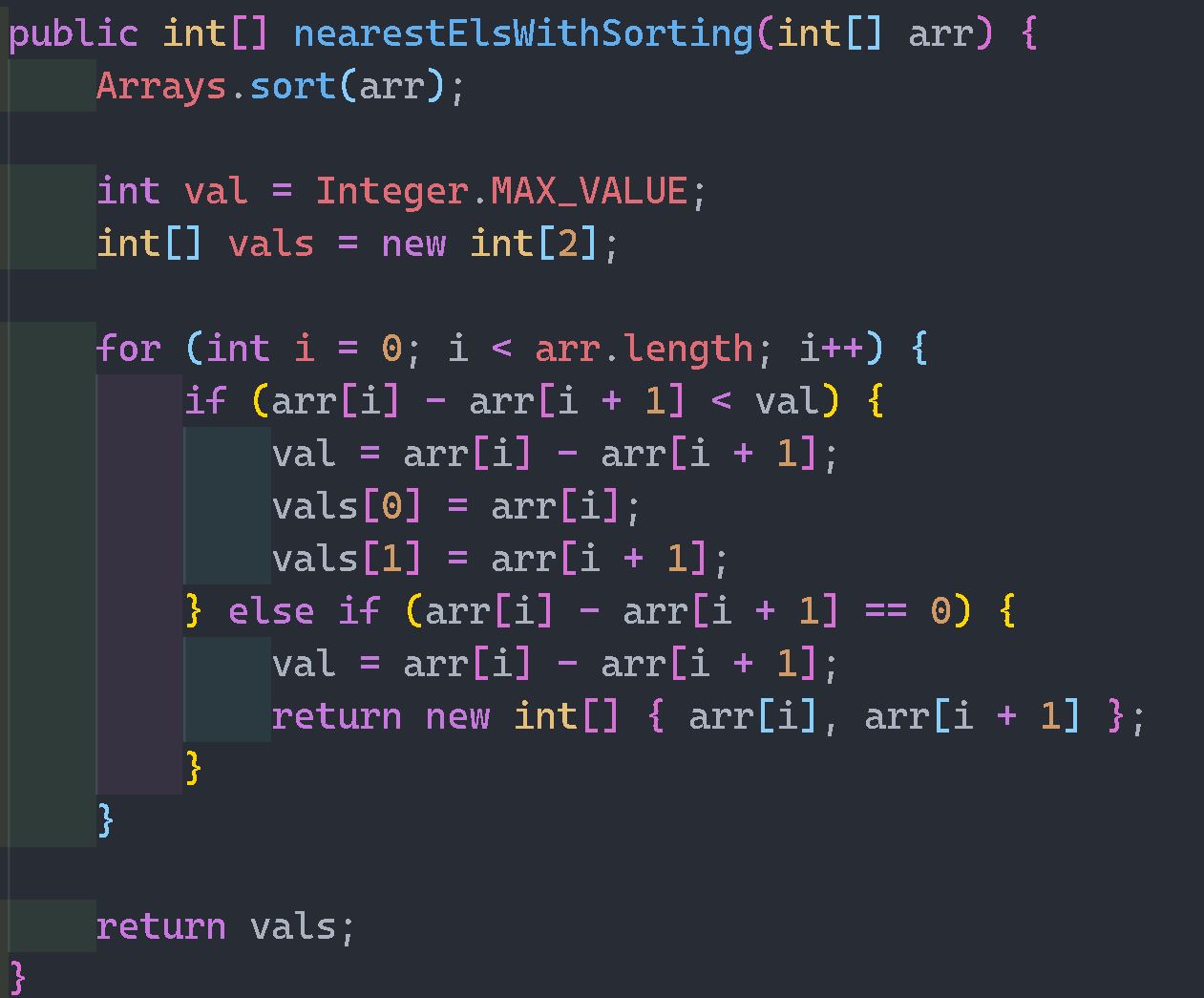
Sıralama olmadan çalışan algoritma sıralama yapılan algoritmadan daha hızlı. Çünkü iki algoritma arasındaki tek fark ikinci algoritmanın sıralanması ve diğer tüm işlemler aynı. Dolayısıyla ikinci algoritmanın çalışma zamanı daha fazla oluyor.

**Sıralama olmayan algoritma** *31.7ms*

**Sıralama olan algoritma** *172.4ms*

1. **Bir Dizideki En Yakın Çifti Bulma**

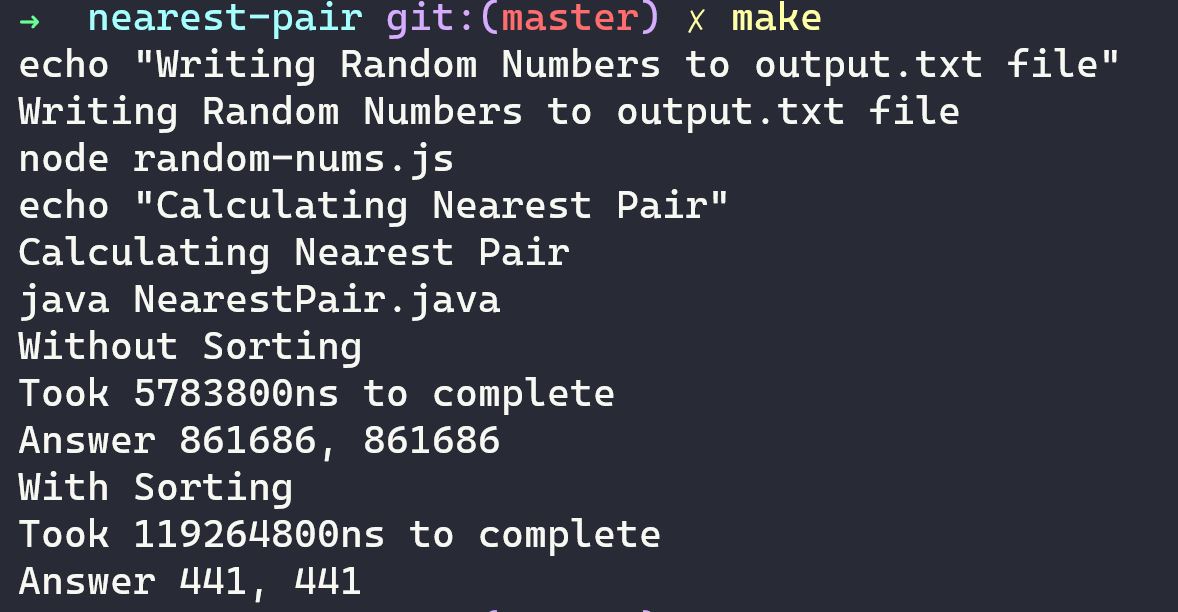
**Şekil 5 - Diziyi sıralamadan dizideki en yakın çifti bulan algoritma.**

****

****

**Şekil 6 - Dizideki en yakın çifti bulan algoritmaların main fonksiyonu.**

**Şekil 7 - Diziyi sıralayarak dizideki en yakın çifti bulan algoritma.**

****

**Şekil 8 - Dizideki en yakın elemanları bulan algoritmaların çalıştırılması ve sonuçlar.**

**Sıralama olmayan algoritma** *5.8ms*

**Sıralama olan algoritma** *119.2ms*