

# DFT - Ayırık Fourier Dönüşümü (Discrete Fourier Transform)

Ses/görüntü işleme veya spektral analiz gerektiren dijital sinyal işleme problemlerinde, ayırık fourier dönüşümü yapmak gereklidir. Bu dönüşümü yapmak için kullanılan en popüler algoritma olan Fast Fourier Transform(FFT) algoritması, dijital örneklerin(samples) sayısının küçük asal çarpanlarına ayrılmasını kullanır. Örnek sayısına  $n$  diyelim. En çok kullanılan FFT algoritması olan Radix-2,  $n$  sayısının  $2$ 'nin tam kuvveti olduğunu varsayar. Bu algoritmanın en büyük eksiği, kullanılabilmesi için örnek sayısının  $n$ 'den büyük, en küçük  $2$ 'nin tam kuvveti olan sayıya artırılması gerekliliğidir. Yani bize  $n$  adet örnek verildiğinde, örnek sayısı  $2n-2$ 'ye kadar artabilir. Bu da bize hemen hemen %100 hesaplama fazlalığı üretir.

Radix-2'ye alternatif olarak Radix-2/3 FFT kullanılabilir. Bu algoritma örnek artırma sayısının üst sınırını gözle görülebilir ölçüde düşürmektedir. Bu algoritmada, örnek sayısı  $n$ , sadece 2 sayısı yerine, 2 ve 3 sayılarının üslerinin çarpımı şeklinde ifade edilmelidir.

$$C_{2,3} = \{n=2^i 3^j \mid i \text{ ve } j \text{ doğal sayı}\}$$

Sizden istenen, bir pozitif  $m$  tamsayısı verildiğinde,  $C_{2,3}$ 'ün içinde bulunan  $m$ 'den büyük veya  $m$ 'e eşit en küçük  $n$  tam sayısını bulmanız.

## **Input**

Input birden fazla testten oluşur.  
Her test için bir adet  $m$  sayısı verilir.  
Inputun sonu  $m$  için 0 verilmesiyle belli edilir.  
 $1 \leq m \leq 2^{31}$   
 $1 \leq \text{TEST\_SAYISI} \leq 2050$

## **Output**

0 olmayan her input  $m$  için,  $C_{2,3}$  kümesine dahil  $m$ 'den büyük eşit, en küçük  $n$  sayısını ekrana bastırınız.

## **Örnek Input**

3  
5  
7  
9  
33  
123  
0

## **Örnek Output**

3  
6  
8  
9  
36  
128