

1. Soru

a) Doğrudur. Big O üst sınırı belirtir, $f(n) \leq c \cdot g(n)$. 3^n her zaman 2^n 'den daha hızlı büyür.

b) Doğrudur. Ω alt sınırı belirtir, $f(n) \geq c \cdot g(n)$. $n \log n^4 = 4n \log n$.
 $4n \log n > n \log n$ olduğu için bu ifade doğrudur.

c) Yanlıştır. Θ hem alt hem üst sınırı belirtir, $c_1 g(n) \leq f(n) \leq c_2 g(n)$
 ve $\lim_{n \rightarrow \infty} \frac{n^2 + n}{n^3}$ durumunda payda daha hızlı büyür bu yüzden yanlıştır.

d) Yanlıştır. Alt sınırı n^8 ve üst sınırı n^9 olmalıdır. ve
 $\frac{0.00000001n^9 + \dots}{n^3} =$ durumunda pay daha hızlı büyür.

2. Soru

$$21253080 \rightarrow 1=2 \text{ olur. } T(n) = 3T(n-1), T(1) = 4$$

$$T(2) = 3T(1) = 3 \cdot 4 = 12 \quad \rightarrow \times 3$$

$$T(3) = 3T(2) = 3 \cdot 3 \cdot 4 = 36 \quad \rightarrow \times 3$$

$$T(4) = 3T(3) = 3 \cdot 3 \cdot 3 \cdot 4 = 108$$

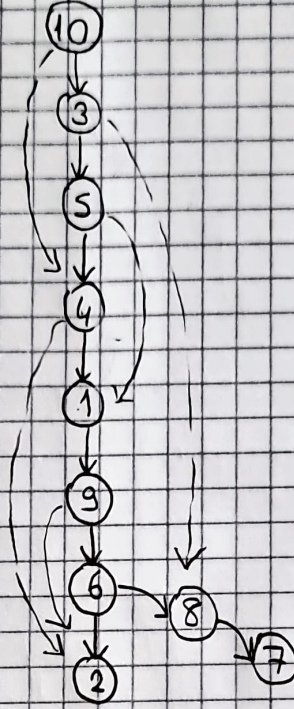
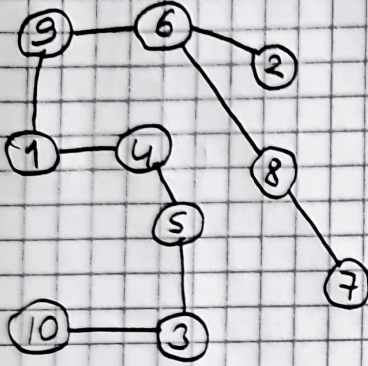
$$T(n) = 3T(n-1) = 3 \cdot 3T(n-2) = 3 \cdot 3 \cdot 3T(n-3) = \dots = 3^{n-1} \cdot \underbrace{T(1)}_4$$

$$T(n) = 3^{n-1} \cdot 4 \quad | \quad 3^{n-1} \cdot 4 \in \Theta(3^{n-1})$$

3. Soru

 $n = \text{uzunluk Kova}$ $\text{solbosKova} = 0$ $\text{sagdolukova} = n$ $i = 1$ $j = n$ $\text{while } i < j$ $\text{if kova}[i] = \text{"doluk"} \text{ And } \text{kova}[j] = \text{"bos"} \text{ then}$ $\text{kova}[i], \text{kova}[j] = \text{kova}[j], \text{kova}[i] \text{ // kovalar yer de\u0131irir.}$ $\text{solbosKova} = \text{solbosKova} + 1$ $\text{sagdolukova} = \text{sagdolukova} - 1$ $i = i + 1$ $j = j - 1$ $\text{else if kova}[i] = \text{"bos"}$ $\text{solbosKova} = \text{solbosKova} + 1$ $i = i + 1$ $\text{else if kova}[j] = \text{"doluk"}$ $\text{sagdolukova} = \text{sagdolukova} - 1$ $j = j - 1$ else $i = i + 1$ $j = j - 1$ end while

Soru 4.

01253080 \rightarrow 10

2 = 20man

2₁: 10 eklenir. stack[10]2₂: 3, 4 eklenir. stack[10, 3, 4]2₃: 4 çıkartılır. stack[10, 3]2₄: 5, 8 eklenir. stack[10, 3, 5, 8]2₅: 8 çıkartılır. stack[10, 3, 5]2₆: 4 eklenir. stack[10, 3, 5, 4]2₇: 1, 2 eklenir. s[10, 3, 5, 4, 1, 2]2₈: 2 çıkartılır. s[10, 3, 5, 4, 1]2₉: 9 eklenir. s[10, 3, 5, 4, 1, 9]2₁₀: 6 eklenir. s[10, 3, 5, 4, 1, 9, 6]2₁₁: 2 eklenir. s[10, 3, 5, 4, 1, 9, 6, 2]2₁₂: 8 eklenir. s[10, 3, 5, 4, 1, 9, 6, 2, 8]2₁₃: 7 eklenir.

stack[10, 3, 5, 4, 1, 9, 6, 2, 8, 7]