Big O

-> Upper Bound/En kötü ihtimal

Big Omega Ω

-> lower Bound / En iyi ihtimal

Big theta Θ thight Bound
Average Case

```
for i in range(len(arr))
    if arr[i] == target
        return i
```

Basit arama

$O(n)$ -> En kötü durum hepsine bakılır

$\Omega(1)$ -> en iyi durum ilkinde

$\Theta(n)$ -> ortalama süre $n/2$ olarak notasyon karşılığı (n)

Binary Search

↳ $O(\log n)$ -> yarıya bölerek aradı
En kötü ihtimalle son arama

↳ $\Omega(1)$ -> En iyi ihtimalle ilk te

↳ $\Theta(\log n)$ -> genelde / ortalamada logn gibi olacak

1.3 State the asymptotic running time in Θ-notation of each of the following algorithms.

```
ALG1(n)
c = 0
for i = 1 to ⌊n/3⌋ do
  c = c + 1
end for
for j = 1 to ⌊n/5⌋ do
  c = c + 1
end for
```

```
ALG2(n)
if n ≤ 1 then
  return 1
else
  return 1+ALG2(n − 2)
end if
```

```
ALG3(n)
for j = 1 to n do
  i = n
  while i ≥ 3 do
    i = ⌊i/2⌋
  end while
end for
```

Solution: Θ(2n)

Solution: Θ(1)

Solution: Θ(n * log2(n))

Maths

$T(n) \rightarrow$ n girdi için zaman masrafı

$n_0 \rightarrow$ Big O notationun anlamlı olacağı min girdi

$c \rightarrow$ constant factor dominant terimin kat sayısı, teoride yok, pratikte etkin

↳ $3n^2 \rightarrow O(n^2)$

↳ c (tüm func sadece $3n^2$ ise)

$f(n) \rightarrow$ dominat terim

$3n^2+4n$

$\downarrow$

$f(n) = n^2$

Slayt 28/29 Big O

$T(n) = \underbrace{3n^2}+5n+2$

$T(n)$ tüm detaylar.

$3n^2$ olayları etkiler

$f(n) = n^2$

$C = ? \rightarrow Cn^2 \approx 3n^2+5n+2$ olmalı

Bu durumda $C=4$ olabilir.

n_0'dan itibaren $(n \geq n_0)$

(Big O) $\rightarrow$ $T(n) \leq c \cdot f(n)$ olmalı

Let $T(n) = 3n^2 + 5n + 2$, and you hypothesize $T(n) = O(n^2)$.

1. Set $f(n) = n^2$.
2. Choose $C = 4$ so that $3n^2 + 5n + 2 \leq 4n^2$.
3. Test where $3n^2 + 5n + 2 \leq 4n^2$. Simplify to $5n + 2 \leq n^2$.
4. For $n \geq 6$, this inequality holds, so you can set $n_0 = 6$.

Thus, $C = 4$ and $n_0 = 6$ work to prove $T(n) = O(n^2)$.

Big O kötü ihtimal olduğundan

$T(n) \leq C \cdot f(n)$ → tahmin daha kötümser.

↑ gerçek süre

↓ Bizim tahmin

Ama Big Ω iyimser olmalı

$T(n) \geq c \cdot f(n)$ olur.

$$f(n) = 3n^2 + 7n + 1 \text{ is } \Omega(n^2).$$

To show $f(n) = \Omega(n^2)$, we need constants C and n_0 such that:

$$f(n) \geq C \cdot n^2 \quad \text{for all } n \geq n_0.$$

1. **Set up the inequality:** Starting with $f(n) = 3n^2 + 7n + 1$, we want:

$$3n^2 + 7n + 1 \geq C \cdot n^2.$$

2. **Rearrange terms:** To isolate terms involving n, we can subtract $C \cdot n^2$ from both sides:

$$3n^2 + 7n + 1 \geq C \cdot n^2 \implies (3 - C)n^2 + 7n + 1 \geq 0.$$

3. **Choose a value for C:** To ensure $(3 - C)n^2 + 7n + 1 \geq 0$ for large n, we need $3 - C > 0$, which implies $C < 3$. So let's try $C = 2$.

   Substituting $C = 2$, the inequality becomes:

$$(3 - 2)n^2 + 7n + 1 \geq 0 \implies n^2 + 7n + 1 \geq 0.$$

4. **Check if the inequality holds:** The inequality $n^2 + 7n + 1 \geq 0$ holds for all $n \geq 0$.
5. **Conclusion:** With $C = 2$ and $n_0 = 0$, we have shown that $f(n) = \Omega(n^2)$.

1. adımda gerçek süre $\geq$ tahmin dedik

Big theta da 2 tahmin ile alt üst sınır Belirlenir.

$C_1 \cdot f(n) \leq T(n) < C_2 \cdot f(n) \quad \forall n > n_0$

Function: $f(n) = 5n^3 - 2n^2 + n$

We want to show that $f(n) = \Theta(n^3)$.

1. **Set up the inequalities:** To show $f(n) = \Theta(n^3)$, we need constants C_1, C_2, and n_0 such that:

$$C_1 \cdot n^3 \leq f(n) \leq C_2 \cdot n^3 \quad \text{for all } n \geq n_0.$$

2. **Determine C_1, C_2, and n_0:**
   - Since $f(n) = 5n^3 - 2n^2 + n$, the dominant term is $5n^3$.
   - For the upper bound, choose $C_2 = 6$ so that:

$$5n^3 - 2n^2 + n \leq 6n^3.$$

     Simplifying, we get $-2n^2 + n \leq n^3$, which holds when $n \geq 2$.
   - For the lower bound, choose $C_1 = 4$ so that:

$$4n^3 \leq 5n^3 - 2n^2 + n.$$

     Simplifying, $n^3 \geq 2n^2 - n$, which also holds when $n \geq 2$.

3. **Conclusion:** $f(n) = \Theta(n^3)$ with $C_1 = 4$, $C_2 = 6$, and $n_0 = 2$.