Soru 1)

King's Collage'ta professor olan Alan Turing, gizli bir iş başvurusunda bulunuyor ve görüşmeye gidiyor, görüşmede geçen şu konuşmalar, filmde etkilendiğim en önemli sahnelerden birisi: Komutan Denniston onun cvsini inceleyerek, “Siz bir dehasınız ve daha 23 yaşında benim başlığını bile anlamadığım bir makale yayınlamanız dehalığınızı kanıtlamaz mı?” sorusuna karşılık Turing’in, “Newton binom teoremini 22 yaşında keşfetmişti, Einstein 26 yaşlarındayken dünyayı değiştiren 4 makale yazdı, ben bunların yanında vasat biriyim.” diyerek cevap vermesidir.

Turing, her gece yarısı değişen ve 189.10^{18} ihtimali olan bu decode işlemi için insan gücünün yetersiz kalacağını ve ancak bir makinenin çözebileceğini söyledi ve bir makine oluşturmak için bütçe istedi fakat komutanları buna izin vermedi, Turing ise inatçı ve kararlı yapısı gereği pes etmeyerek Londradaki, üst komutanlara bir mektup yolladı ve istediği bütçeyi aldı üstelik Enigma’yı çözmek için oluşturulan grubunda lideri oldu.

Grup lideri olur olmaz, 2 kişiyi işten kovdu ve bir personel açığı doğdu, bu personel açığını çözmek için izlediği yolsa müthişti. Bir bulmaca oluşturdu ve gazetede yayınladı. Bu bulmacayı çözenlere ise gizli bir iş teklifi adı altında bir mektup gönderdi ve tüm adayları bir salonda toplayarak, 6 dakikalık bir bulmaca daha verdi, çözen 2 kişiyi ise işe aldı. Bunlardan birisi, Turingten hoşlanacak olan Mrs. Clarke idi.

Makinesini oluşturmaya başladı, hatta ona bir ad bile verdi "Christopher"

Ayrıca çok dikkatimi çeken bir sahnede, Turing’e savaş zamanında, makinesini oluşturmak için oldukça fazla para kaynağı sunuluyordu fakat hala bir sonuç alınamaması komutanları kızdırmıştı. Bu sebeple Turing’in hükümeti oyaladığını düşünen komutanlar onu kovmaya geldiklerinde, ondan hiç hoşlanmayan takım arkadaşları, takım ruhunu konuşturup hep beraber "o giderse bizde gideriz" dediler ve komutanlar onlara 1 ay daha verdi.

Oluşan makinede tüm olasılıkları deniyorlardı her sabah, ta ki gece yarısı şifre değişene kadar fakat Christopher tüm olasılıkları bu sürede deneyemiyordu, zaman çok az geliyordu bu sebeple tüm olasılıkları denemek yerine, mesajlarda hep tekrar ettiklerini bildikleri kelimeleri çözmeye çalışmaya başladılar, örneğin her sabah "Çok yaşa Hitler!" kelimesi kesinlikle mesaj olarak geçiyordu, bu gibi tekrar eden kelimeleri çözerek tüm mesajları çözebilirlerdi ve bu yolla Enigma’yı çözdüler.

Fakat Enigmayı çözdüklerini saklamak zorundaydılar çünkü eğer Almanlar bunu öğrenirse, hemen şifreleme algoritmalarını değiştirecekler ve 2 yıllık bu çalışma boşa gidecekti. Bu yüzden Turing, "saldırıları durdurup atlatmaya izin verecek ve istihbaratın ne kadar etkilendiğini belirleyecek bir sistem" geliştirmeyi teklif etti. İstatiksel Analiz yöntemi ile. Bu teklif için ise büyük bir istihbarat merkezi kuruldu.

Bu şekilde bir karar mekanizması oluşturarak bazı olaylara engel oluyor bazılarına ise olmuyorlardı böylece Enigma’yı çözdükleri bir sır olarak kalıyordu ve Almanların şifreyi çözdüklerinden haberleri olmadan, savaşı kazandılar. Elde ettikleri tüm belgeleri ise yakmaya karar verdiler çünkü herkes Enigma’nın kırılması imkansız bir şifreleyici olduğuna inanıyordu ve olası bir durumda tekrar Enigma’yı kullanabilinirdi. Buda çok güzel bir taktikti.

Beni etkileyen son olaysa, Turing’in eşcinsel olması fakat o zamanlarda eşcinselliliğin yasak olmasından dolayı bu ortaya çıkınca hormonal tedavi cezasından sonra intihar etmesidir. Bilgisayarın kurucusu sayılan bir kişinin intihar ederek ölmesi gerçekten çok üzücü bir şey.

Soru 2)

1. If $f(n) = O(n^{\log_b a-\varepsilon})$ for some constant $\varepsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$.
2. If $f(n) = \Theta(n^{\log_b a} \log^k n)$ with1 $k \geq 0$, then $T(n) = \Theta(n^{\log_b a-\varepsilon} \log^{k+1} n)$.
3. If $f(n) = \Omega(n^{\log_b a+\varepsilon}$ with $\varepsilon > 0$, and $f(n)$ satisfies the regularity condition, then $T(n) = \Theta(f(n))$.

Regularity condition: $af(n/b) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large n.

x_1)

a=0.5 b = 2 f(n) = 1/n (a ≥ 1 özelliğini sağlamadığından, master teorem ile çözülemez)

x_2)

a=3

b=4

f(n)= nlogn

$f(n) = \Omega(n^{\log_b a+\varepsilon}) = \Omega(n^{\log_4 3+\varepsilon})$ for $\varepsilon > 0$ and f(n) satisfies the regularity condition, then $T(n) = \Theta(f(n))$

Regularity conditionı test etmemiz lazım,

Regularity condition: $af(n/b) \leq cf(n)$ for $c<1$

$3(n/4)(\log(n/4)) \leq c\, n\log n$ c = ¾ için bu eşitlik sağlanır. Dolayısıyla bu kuralı uygulayabiliriz.

$f(n) = \Omega(n^{\log_b a+\varepsilon})$ olduğu için $T(n) = \Theta(f(n))$, $T(n) = \Theta(n\log n)$

x_3)

a=3

b=3

f(n) = n/2

$f(n) = \Theta(n^{\log_b a} \log^k n) = \Theta(n^{\log_3 3}) = \Theta(n)$, for k = 0,

So, $T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$ formülünden, $T(n) = \Theta(n^{\log_3 3} \log^{0+1} n)$, $T(n) = \Theta(n\log n)$

x_4)

a = 6

b = 3

$f(n) = n^2 \log n$

$f(n) = \Omega(n^{\log_b a+\varepsilon}) = \Omega(n^{\log_3 6+\varepsilon})$ for $\varepsilon > 0$ and f(n) satisfies the regularity condition, then $T(n) = \Theta(f(n))$

Regularity conditionı test etmemiz lazım,

Regularity condition: $af(n/b) \leq cf(n)$ for $c<1$

$6f(n/3) \leq cf(n)$ for $c<1$

$6(n/3)^2\log(n/3) \leq c\, n^2\log n$, c=2/3 için ifade sağlanıyor. Dolayısıyla bu kuralı uygulayabiliriz.

$T(n) = \Theta(f(n))$, $T(n) = \Theta(n^2\log n)$

x_5)

a = 4

b = 2

$f(n) = n/\log n$

$n^{\log_b a} = n^{\log_2 4} = n^2 > f(n)$

$f(n) = O(n^{\log_2 4-\varepsilon})$ for some constant $\varepsilon > 0$, then $T(n) = \Theta(n^{\log_2 4})$, $T(n) = \Theta(n^2)$.

x_6)

$a = 2^n$

b = 2

$f(n) = n^n$

a, constant olmadığından dolayı master teoremi uygulanamaz.

Soru 3)

a) $T(n) = T(n-1) + 2n - 1$, $T(1) = 1$

$T(1) = 1$

$T(2) = 3$

$T(3) = 9$

$T(4) = 16$

$T(5) = 25$

Guess: $T(n) = n^2$

Tümevarım yöntemi ile kanıtlayalım,

1) Base case:
   $T(1) = 1^2 = 1$
2) Inductive Step:
   Assume true for k-1
   $T(k-1) = (k-1)^2$
3) Prove for k

   $T(k) = T(k-1) + 2k - 1$

   $T(k) = (k-1)^2 + 2k - 1$

   $T(k) = k^2 - 2k + 1 + 2k - 1$

   $T(k) = k^2$

   Bu algoritma verilen ifadenin karesini hesaplar.

   $T(n) = n^2$ dir sonuç.

b) Pozitif verilen inputlar için eğer if koşuluna girilirse sadece return yapıldığından herhangi çarpma işlemi yoktur, yani sayisi 0. Else koşuluna girerse, 1 çarpma işlemi yapıldığından, toplam sayı 1 dir. Bu bilgilere göre şu şekilde bir recurrence relation yazabiliriz,

$T(n) = T(n-1) + 1$, $T(1) = 0$

Şimdi bu recurrence relationu çözelim,

$T(1) = 1$

$T(2) = 2$

$T(3) = 3$

$T(4) = 4$

$T(5) = 5$

Guess: $T(n) = n$

Tümevarım yöntemi ile kanıtlayalım,

1) Base case:
   $T(1) = 1$
2) Inductive Step:
   Assume true for k-1
   $T(k-1) = k – 1$
3) Prove for k

   $T(k) = T(k-1) + 1$

   $T(k) = k – 1 + 1$

   $T(k) = k$

$T(n) = n$ ‘dir sonuç

c) Pozitif verilen inputlar için eğer if koşuluna girilirse sadece return yapıldığından herhangi toplama veya çıkartma işlemi yoktur, yani sayisi 0. Else koşuluna girerse, 1 toplama ve 1 çıkartma işlemi yapıldığından toplam sayi 2 dir. Bu bilgilere göre şu şekilde bir recurrence relation yazabiliriz,

$T(n) = T(n-1) + 2 \quad T(1) = 0$

Şimdi bu recurrence relationu çözelim,

$T(1) = 1$

$T(2) = 3$

$T(3) = 5$

$T(4) = 7$

$T(5) = 9$

Guess: $T(n) = 2n – 1$

Tümevarım yöntemi ile kanıtlayalım,

4) Base case:
   $T(1) = 2.1 – 1 = 1$
5) Inductive Step:
   Assume true for k-1
   $T(k-1) = 2(k-1) – 1 = 2k - 3$
6) Prove for k

   $T(k) = T(k-1) + 2$

   $T(k) = 2k – 3 + 2$

   $T(k) = 2k – 1$

$T(n) = 2n – 1$ ‘dir sonuç.

Soru 4) Kodu yazıldı.

Soru 5)

a) logn complexitye sahip kod yazıldı.

b)

Best case: 2 elemanlı liste gelmesi durumudur. Bu iki elemanlı liste için, 0. elemanını ve 1. Elemanını listeye çevirip compareScales fonksiyonunu çağıracaktır. compareScales fonksiyonu $\Theta(1)$ sürede çalışır, 1. Ve 2. Elemanları listeye çevirmek $\Theta(1)$ süre alır. Dolayısıyla best case = $\Theta(1)$

Yada gelen listenin eleman sayısının tek olduğu durumda son 2 elemanından birisinin rotten olma durumudur. Bu durumda compareScales fonksiyonunu çağıracaktır. compareScales fonksiyonu $\Theta(1)$ sürede çalışır ve karşılaştırdığı elemanlardan birisinin rotten olma durumunda direk o elemanın indexini return eder. Böylece durumda best case durumudur ve karmaşıklığı $\Theta(1)$ dir.

Worst case: Listenin tek elemanlı olması ve rotten değerin son 2 elemandan biri olmaması durumudur. Liste tek elemanlı olunca fazladan compareScales çağırılacak ve ardından liste ikiye bölünerek recursion, liste bitene kadar çağırılacaktır. Her çağırımda liste 2 ye bölündüğünden, worst case durumu $\Theta(\log n)$ dir.

Recurrence relationumuzu yazalım bu algorithma için,

$T(n) = T(n/2) + O(1)$

Master teoremden,

$a=1$

$b=2$

If $f(n) = \Theta(n^{\log_b a} \log^k n)$ with1 $k \geq 0$, then $T(n) = \Theta(n^{\log_b a-\varepsilon} \log^{k+1} n)$ 2 kuraldan,

$f(n) = \Theta(n^{\log_2 1} \log^0 n)$, for $k = 0$

$f(n) = \Theta(1)$

So, $T(n) = \Theta(n^{\log_2 1} \log^{0+1} n) \Rightarrow T(n) = \Theta(\log n)$

Soru 6)

a)

T_1)

$T(1) = 4 = 4.3^0$

$T(2) = 4 . 3 = 4.3^1$

$T(3) = 4. 3 . 3 = 4.3^2$

$T(4) = 4. 3 . 3 . 3 = 4.3^3$

$T(5) = 4. 3 . 3 . 3 . 3= 4.3^4$

Guess: $T(n) = 4.3^{n-1}$

Tümevarım yöntemi ile kanıtlayalım,

1) Base case:
   $T(1) = 4.3^{1-1} = 4$
2) Inductive Step:
   Assume true for k-1
   $T(k-1) = 4.3^{k-2}$
3) Prove for k

   $T(k) = 3T(k-1)$

   $T(k) = 3. 4.3^{k-2} = 4.3^{k-1}$

So, $T(n) = 4.3^{n-1}$

T_2)

$T(n) = T(n-1) + n$

$T(n) = T(n-2) + (n-1) + n$

$T(n) = T(n-3) + (n-2) + (n-1) + n$

…

$T(n) = T(0) + 1 + 2 + … + n$

$T(n) = 0 + 1 + 2 + … + n$

$T(n) = \frac{n(n+1)}{2}$

T_3)

$T(n) = T(n/2) + n$ for $n > 1$, $T(1) = 0$

$T(n) = T(n/2^2) + n/2 + n$

$T(n) = T(n/2^3) + n/2^2 + n/2 + n$

…

$T(n) = T(n/2^k) + n/2^{k-1} + n/2^{k-2} + ... + n/2 + n$

$n = 2^k$ için,

$T(n) = T(1) + 2^k /2^{k-1} + 2^k /2^{k-2} + ... + 2^k /2 + 2^k$

$T(n) = 0 + 2^1 + 2^2 + ... + 2^k$

$T(n) = \sum_{i=1}^{k} 2^i$

$T(n) = (\sum_{i=0}^{k-1} 2^i) -1 + 2^k$

$T(2^k) = (1-2^k)/(1-2) + 2^k – 1$

$T(2^k) = 2^k + 2^k – 2$

$T(2^k) = 2^{k+1} - 2$

$T(n) = 2n - 2$

b)

T_1)

$T(n) = 6T(n-1) – 9T(n-2)$, $T(0) = 1$, $T(1) = 6$

$\alpha^2 - 6\alpha + 9 = 0$

$(\alpha – 3)^2 = 0$

$\alpha = +3$ ve ya $\alpha = -3$

$T(n) = c_1\alpha^n + c_2 n\alpha^n$

$T(0) = 1 = c_1$

$T(1) = 6 = \alpha(1 + c_2)$

1.durum

$\alpha = 3$

$c_2 = 1$

$T(n) = 3^n + n3^n$

$T(0) = 1$, $T(1) = 6$, $T(2) = 27$

2. durum

$\alpha = -3$

$c_2 = -3$

$T(n) = (-3)^n + -3n(-3)^n$

$T(2) = 27 \neq -45$, sağlamadı.

Dolayısıyla çözüm, $T(n) = 3^n + n3^n$ dir.

T_2)

$T(n) = 5T(n-1) - 6T(n-2) + 7^n$

$x^2 - 5x + 6 = 0$

$(x-3)(x-2)$

$x = 3$

$x = 2$

$a_h = \alpha 2^n + \beta 3^n$

$a_p = 7^n$

Guess: $T(n) = A7^n$

$A7^{n+2} - 5\,A7^{n+1} + 6\,A7^n = A7^n$

$49A - 35A + 6A = 1$

$A = 1/20$

$a = a_p + a_h$

$a = \alpha 2^n + \beta 3^n + 1/20(7^n)$ olarak bulunur çözüm.