

TÜRKİYE CUMHURİYETİ
YILDIZ TEKNİK ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



ALGORİTMA ANALİZİ İKİNCİ ÖDEV RAPORU

Öğrenci No: 20011023
Öğrenci Adı Soyadı: Mehmet Alperen Ölçer
Öğrenci E-Posta: alperen.olcer@std.yildiz.edu.tr

Ders/Grup: BLM3021 Algoritma Analizi/ 1.Grup

Ders Yürütücüsü
PROF DR MİNE ELİF KARSLIGİL
6 Kasım 2022

YÖNTEM:

Verilen problemde bir dizideki yan yana olan elemanlar toplamı en büyük olan alt dizinin bulunması istenmiştir.

Brute force yaklaşımında iç içe iki for döngüsü ile her bir eleman için kendinden sonraki elemanlar ile en büyük toplam değeri hesaplanarak bulunmuştur.

BRUTE-FORCE PSEUDO KOD:

```
brute_force ( array, start_index, stop_index, n)
```

```
    max_gain = MINUMUM
```

```
    For i <- 0 to n:
```

```
        sum = 0
```

```
        For j <- i to n:
```

```
            sum += array[j]
```

```
            if sum > max_gain:
```

```
                max_gain = sum
```

```
                range_L = i
```

```
                range_R= j
```

```
    return range_L, range_R , max_gain
```

Divide and Conquer yaklaşımında ise dizi her seferinde ortadaki elemana göre parçalanmıştır. Dizinin sağ tarafında ortadaki elemanı içeren ve dizinin en fazla kazancını içeren alt dizi bulunmuştur. Sol tarafında da aynı işlem tekrarlanarak ortak bir dizi elde edilmiştir. Daha sonra parçalanmış dizinin sağından, solundan ve iki tarafı da kapsayan diziden elde edilen sonuçlar kıyaslanarak en büyük aralık bulunmuştur.

DIVIDE AND CONQUER PSEUDO KOD:

```
find_dMax(array, left, right)
```

```
    N = right-left+1
```

```
    mid = N tekse (N-1)/2, N çiftse ortadaki ikiliden büyük olanın indexi
```

```
    gain = max = array[mid]
```

```
    For j <- mid+1 to right:
```

```
        gain += array[j]
```

```
        if gain > max:
```

```
            max = gain
```

```

        range_R = j
    gain = max
    For j <- mid-1 to left:
        gain += array[j]
        if gain > max:
            max = gain
            range_L = j;
    return range_L, range_R, max

```

solution (array, left, right)

```

if right == left:
    brute_force (array, left, right)
left_gain = solution(array, left, (right + left) / 2)
right_gain = solution( array, ((right + left) / 2) + 1, right)
mid_gain = find_dMax(left_gain.left, right_gain.right)
return max(left_gain, right_gain, mid_gain)

```

BRUTE FORCE ANALYSIS:

Basic operation: for $i=0; i < n$
for $j=i; i < n$

$$T(n) = \sum_{i=0}^n \sum_{j=i}^n 5 = \frac{5 \cdot n(n+1)}{2} \sim n^2$$

$O(n^2), \Theta(n^2), \Omega(n^2)$
ile gösterilebilir.

DIVIDE AND CONQUER:

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

Master Theorem:

$$a=2 \quad b=2 \quad d=1$$

$$a = b^d \rightarrow T(n) = n \log n$$

UYGULAMA:

Pozitif negatif sayı havuzu için çalışmaktadır.

```
Checking correctness
-----
ARRAY : -18 -100 81 -90 65 54 -59 -48 -90 -68 -23 -87 1 48 63 44 53 -21 -21 -35 16 -34 30 -9 -34 75 -98 -70 24 54 -61 58 6 72 69 -29 27 -90 -25 -11 -57 52 2 44
-48 -83 -59 -94 -3 -28 -29 65 38 -98 -44 -96 77 -42 -65 -47 64 -74 -89 -30 98 80 41 -23 43 -83 66 86 21 -79 -18 74 90 23 80 -13 95 -97 -96 33 -95 -88 -10 34 -
77 -23 39 87 3 51 -42 -47 -17 51 31 -74 68 -51 64 -58 -30 99 68 -39 -26 0 0 -30 3 -43 55 -39 -31 -55 47 -8 -26 -13 -20 77 -62 -10 -69 -27 -59 14 -48 -90 63 -84
-96 86 67 72 99 -58 -28 -49 -36 27 8 -81 88 78 -83 36 -30 -9 -77 2 -79 -87 92 -96 -14 -66 18 38 96 -19 7 52 -81 -26 -24 -82 -32 48 -78 32 -73 30 4 -84 8 21 4
31 64 -21 85 85 -8 30 89 78 64 7 -31 12 41 28 -36 12 54 40 -17 -77 -60 5 55 -81 -65 -41 -65 -4 32 -9 79 -3 70 16 -66 -38 -54 -76 93 62 -17 -86 -74 76 42 -10 -5
9 -4 82 24 -29 74 -71 79 94 -84 90 81 -36 23 -75 95 72 95 12 -94 -90 10 82 55 -27 -82 69 -1 46 -37 42 87 11 -24 11 83 -97 -8 62 97 -39 52 30 25 -73 55 -27 99 3
85 58 13 -53 -8 -32 20 10 89 72 -43 -48 -34 96 15 -6 -40 -2 -3 52 -88 -54 65 17 29 -57 44 -16 16 44 87 1 -46 -48 48 46 -28 -79 -91 -39 45 -82 -35 63 -86 81 -4
3 -26 -21 7 79 -56 5 -4 61 -66 39 5 -29 -45 -99 -90 8 -45 -37 -91 54 35 -70 63 97 27 81 -86 -10 47 95 -1 74 -25 -94 5 19 -36 -99 -20 -2 41 37 69 96 39 80 5 46
95 14 -48 30 -4 -85 -21 75 48 -54 -35 -52 -59 16 -78 -32 -25 -21 87 39 -20 -81 -63 73 9 59 22 -100 39 -73 46 -14 -7 99 68 41 66 -100 16 -33 -54 33 15 -13 -51 -
11 56 24 20 -5 -85 -100 -33 -47 26 76 -88 -52 28 3 -73 -26 41 -28 -75 9 -87 44 9 -19 63 55 66 -22 95 67 19 3 -56 -61 98 -41 91 -83 12 17 45 -24 65 -27 31 92 0
-28 -36 -23 82 -71 -79 -57 -38 84 51 -72 -86 46 96 -15 -51 92 24 99 -49 -32 -83 16 85 62 44 3 88 -24 -5 -12 48 12 65 -18 41 39 -22 56 23 -71 84 -10 27 32 75 28
24 -48 27 28 20 -4 -56 -43 -89 88 60 99 -36 -44 -13 65 20 -48 99 -87 91 -23 69 -33 6 6 57 85 -62 -16 13 15 36 93 -57 -44 -11 -61 14 0 -73 -74 -1 -56 -66 38 61
54 91 60 68 34 -62 -11 53 -4 95 -38 -66 86 47 47 -47 -17 40 -4 -8 82 35 58 -18 14 84 34 58 -81 -28 19 25 -85 -68 -7 2 -78 35 -45 -30 30 18 4 -32 -83 -48 21 -4
8 44 69 44 26 -44 -46 -91 71 91 -5 -19 -90 19 53 35 35 85 81 89 -93 16 -56 -23 98 14 34 -33 31 86 40 84 30 10 -20 -91 18 -13 70 -59 -22 -35 23 -60 84 -72 75 -2
9 65 8 -40 24 76 -43 53 27 23 -13 46 -93 -27 -14 91 56 48 -77 65 -81 10 35 -40 88 52 35 -72 -12 63 56 12 28 -84 72 -48 45 81 -94 -28 -43 -7 70 -36 19 -92 -93 -
73 57 -70 44 76 -7 -69 88 -19 83 24 -38 -77 -13 70 35 68 38 -40 -80 -17 -59 78 7 -2 -76 -23 14 -5 -14 21 22 95 4 -34 23 97 -51 11 -70 -68 -65 -56 -93 23 -86 43
-57 53 -97 -37 88 96 94 -4 -53 -82 25 61 13 -89 -17 87 -94 87 -47 -19 36 -46 93 66 -14 80 -89 -55 -45 77 88 -2 -18 91 62 -29 88 56 67 87 26 -8 48 91 -44 83 78
-86 22 83 -4 -42 -63 -59 77 -25 -79 88 20 -23 17 9 75 -100 -48 -11 -29 92 97 90 79 23 -66 -20 -34 -10 -37 -56 -43 38 -73 5 96 -36 46 25 39 67 -35 -88 96 83 21
24 83 73 -87 -46 66 63 96 -3 -62 -70 77 5 -27 93 49 -18 31 77 87 -21 93 33 -95 33 -48 -30 -3 -99 -47 -82 25 -64 43 38 -58 -39 1 90 59 40 73 -12 -55 -54 -19 94
-20 -88 23 -33 -8 17 52 -3 -98 -96 19 -1 -95 73 -31 30 61 -88 -79 56 -74 -26 46 85 -86 -29 -75 -89 -31 7 58 49 19 -67 -84 -89 50 20 -40 4 25 80 55 -18 5 -76 6
5 -34 89 38 74 -85 -88 -27 -100 79 44 77 42 -86 -16 0 63 -96 -14 32 67 -64 -48 80 -7 29 60 -52 -88 -35 -27 29 -17 -38 -81 -42 -23 -69 31 -71 62 -25 -94 5 -59 -
57 57 -95 -53 -57 37 66 -68 89 46 25 19 -42 25 -17 -25 98 64 59 -88 83 69
Result: 178, 862, 4471
```

Bu sayılar için hesap makinesi çıktısı aynıdır.

Max Sum in a Sequence of Integers

This algorithm finds the largest sum of any sub-sequence of numbers in a sequence of integers (both positive and negative).

```
Enter any sequence of integers (separated by spaces or commas):
95 27 23 12 40 53 27 19 91 20 84 77 95 91 18 33 48 88 24 33 72 12 63 20 12 28 88 72 48 43 81 94 28 43 7 70 30 19 92 93 72 27 70 44 70 7 99 88 19 83 24 38 77 13 70 35 68 38 40 80 11 39 78 7 4 4
76 23 14 5 14 21 22 95 4 34 23 97 51 11 70 68 65 56 93 23 86 43 57 53 97 37 88 96 94 4 53 82 25 61 13 89 17 87 94 87 47 19 36 46 93 66 14 80 89 55 45 77 88 2 18 91 62 29 88 56 67 87 26 8 48 91 2
44 83 78 96 22 83 4 42 63 40 77 25 79 88 79 23 17 9 75 100 48 11 29 87 87 90 79 23 66 30 34 10 37 56 43 38 73 5 96 36 46 35 39 67 35 88 96 83 21 34 83 73 87 48 66 63 96 3 62 70 77 5 27 93 40 18 2
Find Largest Sum of Any Sub-Sequence
```

Largest sum of a sub-sequence: 4471

Sadece negatif sayılarda da çalışmaktadır.

```
Checking correctness
-----
ARRAY : -29 -24 -88 -98 -19 -21 -99 -44 -29 -86 -25 -64 -37 -29 -15
Result: 14, 14, -15
```

Sadece pozitif sayılarda da çalışmaktadır.

```
Checking correctness
-----
ARRAY : 59 53 54 58 50 55 56 55 50 52 56 54 53 50 59
Result: 0, 14, 814
```

Az sayılı bir örnekte daha rahat incelemek gerekirse; rekürsif olarak parçalanmış dizi ilk önce 0-0 da 8 sayısını cevap olarak buluyor. Sonrasında öteki parçalarla birleşirken yapılan kıyaslamalarda 8 den daha büyüğü bulunmadığı için cevap 8 olarak kalıyor.

```
Checking correctness
-----
ARRAY : 8 -3 -10 -1 3 -5 8 -10 6 -8
Result: 0, 0, 8
```

Zaman karmaşıklığı tablosu algoritma farklı sayıda inputlarda iki yöntem ile de çalıştırılınca elde edilmiştir.

Test cases for time complexity					

Size: 10 ¹	Result:	7,	7,	40	Time Recursive: 0.000001
Size: 10 ¹	Result:	7,	7,	40	Time Brute Force: 0.000001
Size: 10 ²	Result:	63,	63,	97	Time Recursive: 0.000005
Size: 10 ²	Result:	63,	63,	97	Time Brute Force: 0.000013
Size: 10 ³	Result:	40,	42,	223	Time Recursive: 0.000051
Size: 10 ³	Result:	40,	42,	223	Time Brute Force: 0.000836
Size: 10 ⁴	Result:	4342,	4344,	223	Time Recursive: 0.000458
Size: 10 ⁴	Result:	4342,	4344,	223	Time Brute Force: 0.064741
Size: 10 ⁵	Result:	50715,	50719,	334	Time Recursive: 0.004613
Size: 10 ⁵	Result:	50715,	50719,	334	Time Brute Force: 6.041248
Size: 10 ⁶	Result:	8539,	8543,	339	Time Recursive: 0.043164
Size: 10 ⁶	Result:	8539,	8543,	339	Time Brute Force: 599.567534

VIDEO LINK : <https://youtu.be/0253ADrQg7A>