שעור 3 המשך מציאת שני איברי המערך כך שההפרש ביניהם יהיה מקסימאלי והאיבר הקטן נמצא במערך לפני האיבר הגדול.

בהינתן מערך $a[n]$ של מספרים יש למצוא שני איברים $a[i]$ ו- $a[j]$ כך שההפרש

$a[j] - a[i]$ יהיה מקסימאלי כאשר $j > i$.

דוגמה 1:

קלט:

indexes	0	1	2	3	4	5	6
A[ ]	2	3	1	7	3	9	5

פלט: $i = 2, j = 5$; $a[5] - a[2] = 9 - 1 = 8$, הפרש מקסימאלי הוא 8.

דוגמה 2:

קלט:

indexes	0	1	2	3	4	5	6	7
A[ ]	22	2	12	4	7	3	19	5

פלט: $i = 1, j = 6$; $a[6] - a[1] = 19 - 2 = 17$, הפרש מקסימאלי הוא 17.

חיפוש שלם:

עוברים על כל קטעי המערך $[i, j]$ ומחשבים את ההפרש $a[j] - a[i]$ המקסימאלי.
סיבוכיות האלגוריתם $O(n^2)$.

שימוש בחיפוש בינארי.

1. מחלקים מערך לשני חלקים שווים
2. ומחשבים את ההפרש המקסימאלי `leftDiff` בחלק השמאלי ואת ההפרש המקסימאלי בחלק הימני של המערך `rightDiff`.
3. מחשבים את האיבר המקסימאלי `maxRight` חלק הימני ואת האיבר המינימאלי `minLeft` בחלק השמאלי.
4. ההפרש המקסימאלי של המערך המאוחד הוא

```
maximum(maxRight - minLeft, leftDiff, rightDiff)
```

סיבוכיות האלגוריתם: מספר חלוקות הוא $O(log_2 n)$, חישוב איבר מינימאלי ומקסימאלי $O(n)$, סיבוכיות האלגוריתם היא $O(nlog_2 n)$.

פסדו-קוד

```
int maxDiff(a[], left, right)
      if(left >= right) return -∞
      mid = (right+left)/2
      leftDiff = maxDiff(a, left, mid)
      rightDiff = maxDiff(a, mid+1, right)
      minLeft = min(a[left, mid])
      maxRight = max(a[mid, right])
      int diff = maxRight - minLeft
      return maximum(maxRight - minLeft, leftDiff, rightDiff)
end-maxDiff
```

שימוש ב-best

אלגוריתם.

1. בונים מערך עזר $t[n-1]$ המכיל מהפרשים של האיברים הסמוכים של המערך המקורי:

$$t[i] = a[i] - a[i-1], i = 1,2, \dots, n-1$$

2. מפעילים **best** על המערך הדש ומקבלים את ההפרש המקסימאלי בין שני איברי המערך.

סיבוכיות האלגוריתם $O(n)$.

פסדו-קוד:

```
int bestAndMaxDifference(a[])
      t[n-1]
      for i = 1 to n-1
            t[i-1] = a[i] - a[i-1]
      end-for
      bestSum = best(t)
      return bestSum
end-bestAndMaxDifference
```

נכונות האלגוריתם.

יהיה קטע בעל סכום מקסימאלי של מערך t הוא $[p, q]$, כלומר

$$bestSum(t) = t_p + t_{p+1} + \cdots + t_q,$$

אבל

$$besSum(t) = \left(a_{p+1} - a_p\right) + \left(a_{p+2} - a_{p+1}\right) + \cdots +$$
$$\left(a_{q-2} - a_{q-1}\right) + \left(a_q - a_{q-1}\right) = a_q - a_p$$

מכאן נובע כי הסכום של קטע במערך t שווה להפרש בין האיברים הקיצוניים של הקטע.

מסקנה. בהינתן מערך $a[n]$ של מסרים ההפרש המקסימאלי בין שני איברי המערך $a[j] - a[i]$ כאשר $j > i$ שווה לסכום המקסימאלי של איברי מערך t המורכב מהפרשים של איברים סמוכים של המערך המקורי. והקטע שמהווה סכום מקסימאלי של מערך עזר הוא קטע $[i, j]$ שקצבותיה מהווים הפרש מקסימאלי:

$$\max_{i<j}\left(a_j - a_i\right) = best(t)$$

אלגוריתם נוסף לחישוב ההפרש המקסימאלי.

1. עוברים על המערך משמאל לימין.
2. שומרים על שני משתנים :
   maxDiff - ההפרש המקסימאלי (זו תהיה התשובה הסופית שלנו),
   min – האיבר המינימאלי שאנו נתקלים בו כאשר עוברים על המערך.
3. כאשר ההפרש בין **a[i]** לבן האיבר המינימאלי גדול מ-**maxDiff** מעדכנים את **maxDiff**
4. כאשר **a[i]** קטן מהאיבר המינימאלי – מעדכנים את האיבר המינימאלי.

פסדם-קוד:

```
int maxDifferrence(a[])
     maxDiff = a[1] - a[0]
     min = a[0]
     for i = 1 to < n-1
          if (a[i] - min > maxDiff)
               maxDiff = a[i] - min
          if (a[i] < min)
               min = a[i]
     end-for
     return maxDiff
end-maxDifferrence
```

כאשר המערך ממוין בסדר יורד הפונקציה מחזירה מספר שלילי.