```
                    array
                      ↑       low    high    input۰ Array arr, L, H    ①
        f(arr, L, H){                        output۰ max۰sum

            av = [ L+H ]  → میانگین
                   [ ‾‾ ]
                   [  ۲ ]
            if(L ==H)                         ← اول function
                return arr[L]

            Sum = f( arr, L, av)
            if( sum  <  f(arr, av+۱, H))       ۱. پارتیشن سمت راست
                sum = f(arr, av+۱, H)          ← تشکیل میده
            if( sum < f2 ( ᵃʳʳ L, av, H))
                          ۰۰
                sum = f2 (arr, L, av, H)

        return Sum

    y

                                              NegInfinit = -∞
                                              Pos Infinit = +∞
        f2( arr, low, average, high)

            Sum = 0
            for index = average to Low          ← دو function
                Sum = Sum+ arr [index]
            if        if (Sum) Neg Infinit)
                        Neg Infinit = Sum


            Sum = 0
            for index = averag +1 to high
                sum = Sum+ arr [index]
                if( sum > Pos Infinit )
                pos Infinit = Sum

            result = Neg Infinit + pos Infinit
            if(result < Neg Infinit)
                result = Neg Infinit
            if( result < Pos Infinit)
                result = pos Infint

    return result
```

② $f(arr, L, H)$

ورودی) B آرایه arr ، که در Low ، $High$

خروجی) B جایی که نقطه‌ی مرتفع نری دارد ← $[res]$ در آرایه
علی قادرگیبی

```
f(arr, L, H) {
    av = ⌊ L+H / ۲ ⌋          ← میانگین
    if( arr[av] > arr[av-1] && arr[av] > arr[av+1])
        return av
    else if (arr[av] > arr[av+1] && arr[av] < arr[av-1])
        return f(arr, L, av)     ⟵ فراخوانی بازگشتی
    else
        return f(arr, av, H)
```

هربار نصف آرایه فراخوانی می‌شه

اگر دقت کنیم آرایه‌مون با هر کشش ارتفاع درصد

$T(n) = T(\frac{n}{۲}) + C$

$\frac{۴}{۲}$ و $logn$ ⟶ $logn \times C = clogn$ ⟶ $\boxed{T(n) = O(logn)}$

---

③ $f(arr, L, H)$ $\underset{L}{\underline{\qquad}} \overset{H}{\underline{\qquad}}$

ورودی) B آرایه arr، که در Low، $High$

خروجی) B کمترین ارتفاع (lowestHeight) در آرایه
علی قادرگیبی

$\boxed{\text{function} \quad f(arr, L, H) }$ {

    $av = ⌊ \dfrac{L+H}{۲} ⌋$ ⟶ میانگین

    $point1 = -1$
    $point2 = -1$

①   if (arr[av] < arr[av+1] && arr[av] < arr[av-1])
        return av

②   if(H - L < ۲) ⟶ تعداد اعضا <=2 ⟶ $\boxed{-1}$
        return -1   برای گردانیم و مسئله جواب نداره.

     point1 = f(arr, L, av)   در کل مسئله ۴ حالت داره

③   if(point1 != -1)
        return point1
     point2 = f(arr, av, H)

④   if(point2 != -1)
        return point2

return -1 }

$f(arr, L, H)$ ④

دردو T آرایه ی arr و دو سر Low و $High$

خب؟ چه یک ویژگی ای کدنک متفاوت دراربا دررو؟

```
findCoin (arr, L, H){
    if( L == H)
        return arr[L]
    if(  Σ arr[i] == Σ        Count2
        i=L          i=Count1+1
        return findCoin (arr, count1 +1, H)
    elseif( Σ arr[i] == Σ arr[i] )
           i=L        i=Count1+1
        return findCoin (arr, count1+1, count2)

    else
        return find coin (arr, L, count1)
```

$$Count1 = \frac{n}{2}$$
$$Count2 = \frac{n \cdot 1}{2}$$

① تسیم به دو دسته ② اگر آرایه ۱ و ۲ باهم مساوی نباشد ③ ترکیب کای دسته ی کوچک تر

④ متوقف تا جایی که؛ دسته ی تک ماند

هست را به دای کوچک

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

Time complexity $\Theta \boxed{\log_2 n}$

$f(arr\, L, H)$ ل

~~find Answer f~~

ورودی 8 (آرایه ی arr و آدرس های Low, High)

خروجی 8 آدرس (مجموعه ی) آدرس سه

find Asnwer (arr , L , H) {

   if (H == L)
     return   arr [H] ,

                                $T(n) = 2T\left(\frac{n}{2}\right) + Cn$

   $av = \lfloor \frac{L+H}{2} \rfloor$

                                $T(n) = O(n \log n)$

   point1 = findAswer (arr , L , av)

   point2 = find Answer (arr , av+1 , H)

      n = point2 [0]

   for (i = L   to   av)
       if (~~ali~~ m (i) < m (n) && y(i) < y(n))
         Delete point2 [i]

                                  → اجتماع دو

   return (point1 ∪ point2)
                         S