最大字段和问题：

问题描述：

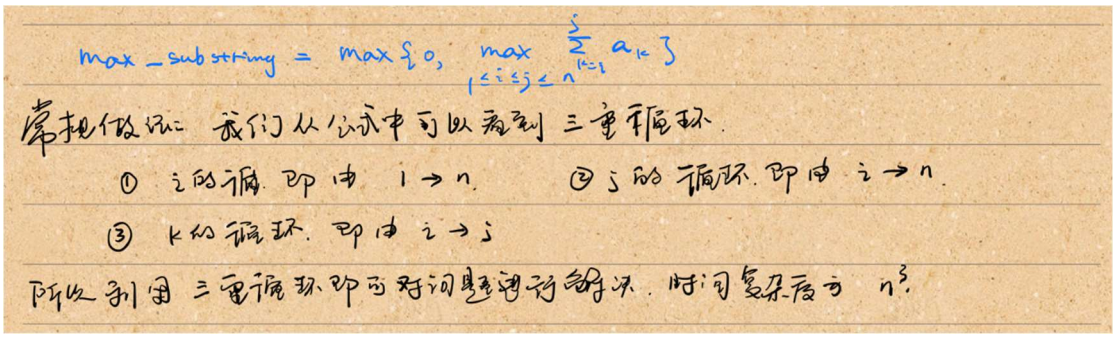
给定由 n 个整数组成的序列 a1，a2，……，an ，求该序列字段和的最大值，当所以整数均为负值时定义其最大字段和为 0。数学描述为：

$$MAX\_SUBSTRING = MAX\{0, \max_{1<=i<=j<=n} \sum_{k=i}^{j} a[k]\}$$

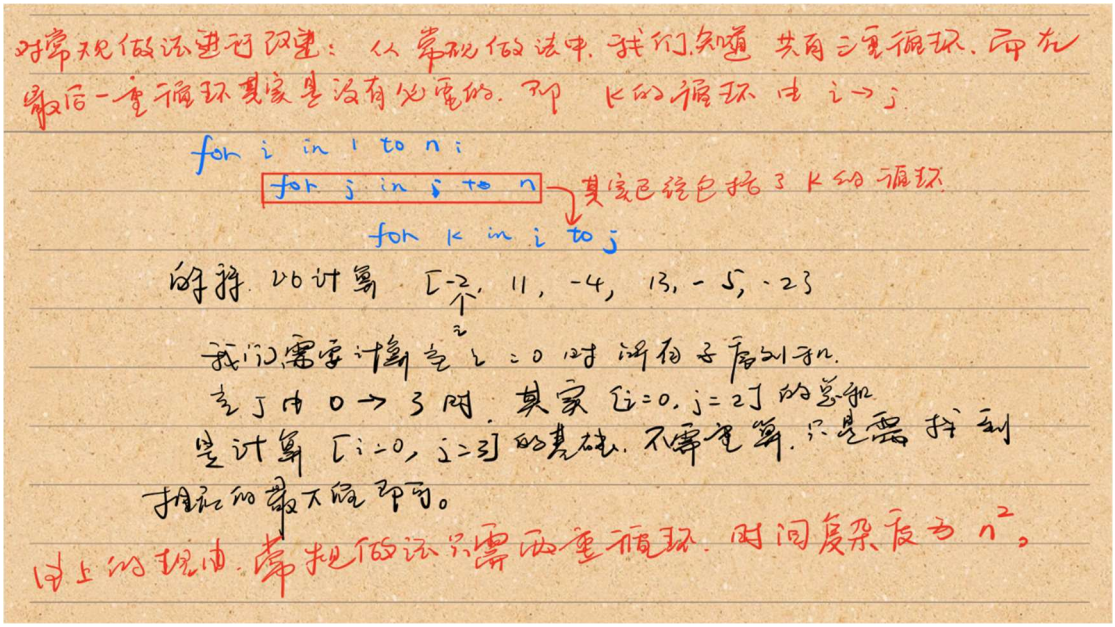
例子：给定{－2，11，－4，13，－5，－2}，

最大字段和为 11+（－4）+13 = 20

常规做法如下：



对常规做法进行修改：



分治法分析:

分治法求解.

将最子段和 $a[1:n]$ 分为 $a[1:n/2]$ 和 $a[n/2+1:n]$ 两段. 则可能有下列三种情况:

1) $a[1:n]$ 的最大子段和 $=a[1:n/2]$ 的最大子段和

2) $a[1:n]$ 的最大子段和 $=a[n/2+1:n]$ 的最大子段和

3) $a[1:n]$ 的最大子段和 $=\sum_{k=i}^{j} a_k$, $1\le i\le n/2$, $n/2+1\le j\le n$

此时, 可分别计算 $a[1:n/2]$ 和 $a[n/2+1:n]$ 的最大子段和.

$$s_1=\max_{1\le i\le n/2}\sum_{k=i}^{n/2} a[k],\quad s_2=\max_{n/2+1\le i\le n}\sum_{k=n/2+1}^{i} a[k]$$

$$max\_substring = s_1+s_2$$

复杂性分析:

$$T(n)=\begin{cases} O(1) & n\le c \\ 2T\left(\frac{n}{2}\right)+O(n) & n>c \end{cases}$$

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

动态规划分析:（算法复杂度为 O（n） ）

$$max\_substring=\max\{0,\ \max_{1\le i\le j\le n}\sum_{k=i}^{j} a_k\}$$

$$\max_{1\le i\le j\le n}\sum_{k=i}^{j} a_k=\max_{1\le j\le n}\max_{1\le i\le j}\sum_{k=i}^{j} a_k \qquad 记\ b_j=\max_{1\le i\le j}\sum_{k=i}^{j} a_k$$

$$\Rightarrow \max_{1\le i\le j\le n}\sum_{k=i}^{j} a_k=\max_{1\le j\le n} b_j$$

因为 $b_j=\max_{1\le i\le j}\sum_{k=i}^{j} a_k \Rightarrow b_{j+1}=\max_{1\le i\le j+1}\sum_{k=i}^{j+1} a_k$

$$=\max_{1\le i\le j}\sum_{k=i}^{j} a_k+a_{j+1}$$

$$=b_j+a_{j+1}$$

即 $b_j>0$ 时 $b_{j+1}=b_j+a_{j+1}$

$b_j=0$ 时, $b_{j+1}=a_{j+1}$

因为: $\max_{1\le i\le j\le n}\sum_{k=i}^{j} a_k=\max_{1\le j\le n} b_j$

$$\Rightarrow b_j=\max_{1\le j\le n}\{b_{j-1}+a_j,\ a_j\}$$

状态-转移方程