最大连续子序列和，非常经典的题。

当我们从头到尾遍历这个数组的时候，对于数组里的一个整数，它有几种选择呢？它只有两种选择： 1、加入之前的SubArray；2. 自己另起一个SubArray。那什么时候会出现这两种情况呢？

如果之前SubArray的总体和大于0的话，我们认为其对后续结果是有贡献的。这种情况下我们选择加入之前的SubArray

如果之前SubArray的总体和为0或者小于0的话，我们认为其对后续结果是没有贡献，甚至是有害的（小于0时）。这种情况下我们选择以这个数字开始，另起一个SubArray。

设状态为{f[j]}，表示以{S[j]}结尾的最大连续子序列和，则状态转移方程如下：

$$\begin{aligned} f[j] &= \max\left\{f[j-1]+S[j], S[j]\right\}, \text{其中 } 1 \leq j \leq n \\ target &= \max\left\{f[j]\right\}, \text{其中 } 1 \leq j \leq n \end{aligned}$$

解释如下：

\item 情况一，S[j]不独立，与前面的某些数组成一个连续子序列，则最大连续子序列和为$f[j-1]+S[j]$。
\item 情况二，S[j]独立划分成为一段，即连续子序列仅包含一个数S[j]，则最大连续子序列和为$S[j]$。

其他思路：

\item 思路2：直接在i到j之间暴力枚举，复杂度是$O(n^3)$
\item 思路3：处理后枚举，连续子序列的和等于两个前缀和之差，复杂度$O(n^2)$。
\item 思路4：分治法，把序列分为两段，分别求最大连续子序列和，然后归并，复杂度$O(n\log n)$
\item 思路5：把思路2$O(n^2)$的代码稍作处理，得到$O(n)$的算法