Given *n* non-negative integers representing the histogram's bar height where the width of each bar is 1, find the area of largest rectangle in the histogram.

  
Above is a histogram where width of each bar is 1, given height = [2,1,5,6,2,3].

  
The largest rectangle is shown in the shaded area, which has area = 10 unit.

**Example:**

**Input:** [2,1,5,6,2,3]

**Output:** 10

问题分析：

其实我们需要计算的是每一个柱子分别往左右延伸直到遇到小于自己的柱子得到的宽度。这个宽度可以分为左右两部分。

我们用栈来存储当前柱子之前的柱子的有效信息。栈内的每一个节点存储柱子的长度，位置index和包括柱子本身在内柱子左边比其高（>=）的柱子的数目，显然该树木>=1。

当访问当前柱子时，将count初始化为1：

1. 循环将柱子的高度和栈顶的柱子的高度比较
   1. 如果柱子的高度>栈顶的柱子的高度，那么挑出循环
   2. 否则
      1. 将栈顶元素出栈，count = count + 栈顶元素的count。
      2. 计算栈顶元素的面积=长度+(栈顶元素的count + 当前位置和栈顶元素位置的差值)，更新最大值
2. 将当前柱子压栈

访问完所有元素之后，最后我们将还留在栈中的元素出栈，计算面积，更新最大值