Largest Rectangular Area in a Histogram

```
from math import ceil, log2
def minVal(hist,x, y) :
   if x==-1:
       return x
   return x if (hist[x] < hist[y]) else y</pre>
def getMid(s, e) :
def RMQUtil( hist,st, ss, se, qs, qe, index):
   if (qs \le ss and qe >= se):
       return st[index]
   if (se < qs or ss > qe):
   mid = getMid(ss, se)
   return minVal(hist,RMQUtil(hist,st, ss, mid, qs,
                        qe, 2 * index + 1),
               RMQUtil(hist,st, mid + 1, se,
                        qs, qe, 2 * index + 2))
def RMQ( hist,st, n, qs, qe) :
   if (qs < 0 \text{ or } qe > n - 1 \text{ or } qs > qe):
       print("Invalid Input")
   return RMQUtil(hist,st, 0, n - 1, qs, qe, 0)
def constructSTUtil(hist, ss, se, st, si):
   if (ss == se):
       st[si] = ss
       return st[si]
  mid = getMid(ss, se)
   return st[si]
```

```
def constructST( hist, n):
  x = (int)(ceil(log2(n)))
  max_size = 2 * (int)(2**x) - 1
   return st
def max_area_histogram(hist):
   area=0
  st = constructST(hist, len(hist))
      def fun(left, right):
           nonlocal area
           if left==right:
           index = RMQ(hist,st, len(hist), left, right-1)
          area=max(area, hist[index] * (right-left))
          fun(index+1, right)
          fun(left,index)
       return (area)
hist = [6, 2, 5, 4, 5, 1, 6]
print("Maximum area is",
  max area histogram(hist))
```