import java.util.Stack;

public class Histogram {

public static int largestRectangleArea(int[] height) {

if (height == null || height.length == 0) {

return 0;

}

Stack<Integer> stack = new Stack<Integer>();

int max = 0;

int i = 0;

while (i < height.length) {

//push index to stack when the current height is larger than the previous one

if (stack.isEmpty() || height[i] >= height[stack.peek()]) {

stack.push(i);

i++;

} else {

//calculate max value when the current height is less than the previous one

int p = stack.pop();

int h = height[p];

int w = stack.isEmpty() ? i : i - stack.peek() - 1;

max = Math.max(h \* w, max);

}

}

while (!stack.isEmpty()) {

int p = stack.pop();

int h = height[p];

int w = stack.isEmpty() ? i : i - stack.peek() - 1;

max = Math.max(h \* w, max);

}

return max;

}

// Driver program to test above function

public static void main(String[] args)

{

int hist[] = { 6, 2, 5, 4, 5, 1, 6 };

System.out.println("Maximum area is " + largestRectangleArea(hist));

}

}