You are given a histogram in the form of a list \( H \) of natural numbers. Write a program to find the longest segment of \( H \) in which the height (each item) is at least as large as the segment length.

Here is a sketch of the solution. The specifications are informal, the proofs are missing, and the timing is missing.

I will check all segments \( m..n \) from longest (\( \#H \)) to shortest (0), and at each length from left to right, stopping the first time I find a square. For each length \( n-m \), there are \( \#H - (n-m) + 1 \) segments to check. For each segment, we can discard it when we find the first height (\( H_i \)) that's too short (< segment length \( n-m \)). A longest segment will be found; the empty segment if nothing longer. So there's no need to check whether we have run out of segments.

\[
S = (m',..n' \text{ is the base of the largest square})
\]
\[
R = (m..n \text{ is the next segment to be checked})
\]
\[
Q = (m \leq i \leq n \text{ and } m..i \text{ is fine and } i..n \text{ needs to be checked})
\]

\[
S \leftarrow m:=0. \quad n:=\#H. \quad R
\]
\[
R \leftarrow i:=m. \quad Q
\]
\[
Q \leftarrow \text{if } i=n \text{ then } \text{ok}
\]
\[
\quad \text{else if } H_i \geq n-m \text{ then } i:=i+1. \quad Q
\]
\[
\quad \text{else if } n<\#H \text{ then } m:=m+1. \quad n:=n+1 \text{ else } n:=\#H-m-1. \quad m:=0 \text{ fi.}
\]
\[
R \text{ fi fi}
\]