```
i = 0      i is running index (inc by 2 every iteration)
while i < length(A)-1
   x = A[i]          # let x and y hold the next to elements in A
   y = A[i+1]

   if x < y then    # ensure that x is not smaller than y
      swap x and y

   j = i - 1          # j is the index used to find the insertion point
   while j >= 0 and A[j] > x          # find the insertion point for x
     A[j+2] = A[j]# shift existing content by 2
     j = j - 1
   end while
   A[j+2] = x      # store x at its insertion place
           A[j+1] is an available space now

   while j >= 0 and A[j] > y          # find the insertion point for y
      A[j+1] = A[j]           # shift existing content by 1
      j = j - 1
   end while
   A[j+1] = y      # store y at its insertion place

   i = i+2
end while

if i = length(A)-1            # if length(A) is odd, an extra
   y = A[i]          # single insertion is needed for
   j = i - 1          # the last element
   while j >= 0 and A[j] > y
      A[j+1] = A[j]
      j = j - 1
   end while
   A[j+1] = y
end if
```