finding lowest card :::::::::::::::

Remember, there are two parts to teaching someone to make french toast: (1) know the algorithm, (2) express the algorithm in a specific vocabulary and grammar.

Compare these two solutions:

Notes: c[2] is shorthand for 'card in spot 2'
      swap n, m is shorthand for 'swap cards in spots n and m'

Algorithm 1:

if c[2] < c[1] then swap 2, 1
if c[3] < c[1] then swap 3, 1
if c[4] < c[1] then swap 4, 1
if c[5] < c[1] then swap 5, 1
if c[6] < c[1] then swap 6, 1
if c[7] < c[1] then swap 7, 1
if c[8] < c[1] then swap 8, 1
if c[9] < c[1] then swap 9, 1
if c[10] < c[1] then swap 10, 1

Algorithm 2:

if c[10] < c[9] then swap 10, 9
if c[9] < c[8] then swap 9, 8
if c[8] < c[7] then swap 8, 7
if c[7] < c[6] then swap 7, 6
if c[5] < c[4] then swap 5, 4
if c[2] < c[1] then swap 2, 1

Are these the same algorithm? If so, why? If not, why not? Do you prefer one to the other? Why or why not?

expressed using variables and repeat :::::::::::::::::