To search for 19 in the list

1 2 3 5 6 7 8 10 12 13 15 16 18 19 20 22,

ALGORITHM 3 The Binary Search Algorithm.

procedure *binary search* (x: integer, $a_1, a_2, \ldots, a_n$: increasing integers)
$i := 1$ {i is left endpoint of search interval}
$j := n$ {j is right endpoint of search interval}
while $i < j$
    $m := \lfloor (i + j)/2 \rfloor$
    if $x > a_m$ **then** $i := m + 1$
    else $j := m$
if $x = a_i$ **then** *location* $:= i$
else *location* $:= 0$
return *location* {*location* is the subscript i of the term a_i equal to x, or 0 if x is not found}

To search for 19 in the list

1 2 3 5 6 7 8 10 12 13 15 16 18 19 20 22,

first split this list, which has 16 terms, into two smaller lists with eight terms each, namely,

1 2 3 5 6 7 8 10 12 13 15 16 18 19 20 22.

Then, compare 19 and the largest term in the first list. Because $10 < 19$, the search for 19 can be restricted to the list containing the 9th through the 16th terms of the original list. Next, split this list, which has eight terms, into the two smaller lists of four terms each, namely,

12 13 15 16 18 19 20 22.

Because $16 < 19$ (comparing 19 with the largest term of the first list) the search is restricted to the second of these lists, which contains the 13th through the 16th terms of the original list. The list 18 19 20 22 is split into two lists, namely,

18 19 20 22.

Because 19 is not greater than the largest term of the first of these two lists, which is also 19, the search is restricted to the first list: 18 19, which contains the 13th and 14th terms of the original list. Next, this list of two terms is split into two lists of one term each: 18 and 19. Because $18 < 19$, the search is restricted to the second list: the list containing the 14th term of the list, which is 19. Now that the search has been narrowed down to one term, a comparison is made, and 19 is located as the 14th term in the original list. ◀

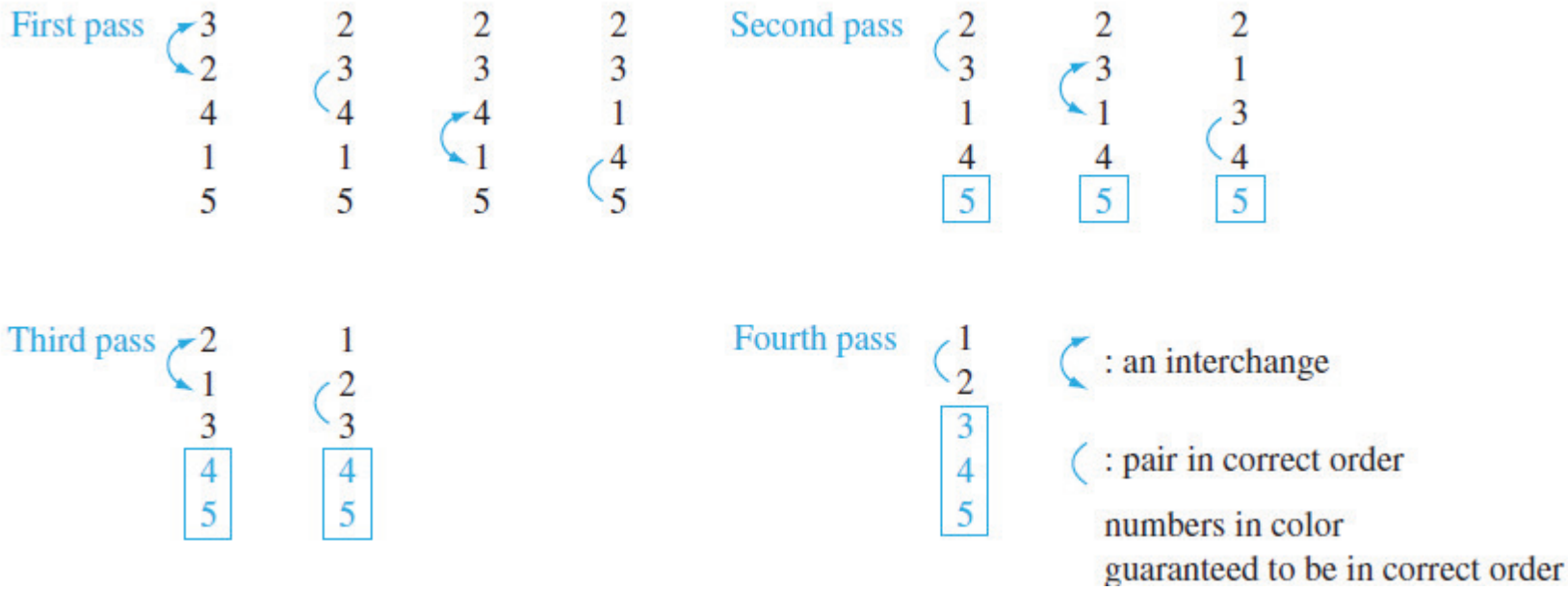
Use the bubble sort to put 3, 2, 4, 1, 5 into increasing order.

ALGORITHM 4 The Bubble Sort.

procedure *bubblesort*($a_1, \ldots, a_n$: real numbers with $n \geq 2$)
for $i := 1$ **to** $n - 1$
    for $j := 1$ **to** $n - i$
        if $a_j > a_{j+1}$ **then** interchange a_j and a_{j+1}
{$a_1, \ldots, a_n$ is in increasing order}

Algorithm Trace

I	1	1	1	1	2	2	2	3	3	4					
J	1	2	3	4	1	2	3	1	2	1					
N	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
n-1	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
n-i	4	4	4	4	3	3	3	2	2	1					
Aj	3	3	4	4	2	3	3	2	2	1					
Aj+1	2	4	1	5	3	1	4	1	3	2					
Aj>aj+1	T	F	T	F	F	T	F	T	F	F					
ai	2 3 4 1 5	2 3 4 1 5	2 3 1 4 5	2 3 1 4 5	2 3 1 4 5	2 1 3 4 5	2 1 3 4 5	1 2 3 4 5	1 2 3 4 5	1 2 3 4 5					


First pass
3 2 4 1 5
2 3 4 1 5
2 3 4 1 5
2 3 1 4 5
Second pass
2 3 1 4 5
2 3 1 4 5
2 1 3 4 5
Third pass
2 1 3 4 5
1 2 3 4 5
Fourth pass
1 2 3 4 5
: an interchange
: pair in correct order
numbers in color
guaranteed to be in correct order