Sort the following number by Quick sort algorithm.
Consider the last element as pivot element.
14,16,7,2,5,9,10,4,20,80,60,50
If the elements are the given in reverse order what will be the complexity of quick sort. Explain with an example

Given:

14	16	7	2	5	9	10	4	20	80	60	50

PASS 1:

1 Initialize Left =0, Right = Max-1 and as given Pivot = last element

14	16	7	2	5	9	10	4	20	80	60	**50**
Left											Right+Pivot

2 Compare, if a[left]<a[pivot]

	a[left]	<	a[pivot]		**Pivot=50**
Left=	14	<	50	TRUE	Left++
Left=	16	<	50	TRUE	Left++
Left=	7	<	50	TRUE	Left++
Left=	2	<	50	TRUE	Left++
Left=	5	<	50	TRUE	Left++
Left=	9	<	50	TRUE	Left++
Left=	10	<	50	TRUE	Left++
Left=	4	<	50	TRUE	Left++
Left=	20	<	50	TRUE	Left++
Left=	**80**	<	50	FALSE	Swap Left and Pivot and Set Pivot = Left

14	16	7	2	5	9	10	4	20	**50**	60	**80**
									Left+Pivot		Right

3 Compare, if a[right]>a[pivot]

	a[right]	>	a[pivot]		**Pivot=50**
Right=	80	>	50	TRUE	Right--
Right=	60	>	50	TRUE	Right--
Right=	50			**Here, right=pivot=50. Terminate the loop and end Pass 1**	
				After Pass 1, Pivot=50 is at its best position	

14	16	7	2	5	9	10	4	20	**50**	60	80
				Sub-array 1						Sub-array 2	

PASS 2: Consider the left Sub-array i.e. sub-array 1

1 Initialize Left =0, Right = Max-1 and as given Pivot = last element

14	16	7	2	5	9	10	4	**20**
Left+Pivot								Right

2 Compare, if a[right]>a[pivot]

	a[right]	>	a[pivot]		**Pivot=14**
Right=	20	>	14	TRUE	Right--
Right=	**4**	>	14	FALSE	Swap Right and Pivot and Set Pivot = Right

4	16	7	2	5	9	10	**14**	20
Left							Right+Pivot	

3 Compare, if a[left]<a[pivot]

	a[left]	<	a[pivot]		**Pivot=14**
Left=	4	<	14	TRUE	Left++
Left=	**16**	<	14	FALSE	Swap Left and Pivot and Set Pivot = Left

4	14	7	2	5	9	10	16	20

Left+Pivot Right

4 Compare, if a[right]>a[pivot]

	a[right]	>	a[pivot]		**Pivot=14**
Right=	16	>	14	TRUE	Right--
Right=	**10**	>	14	FALSE	Swap Right and Pivot and Set Pivot = Right

4	10	7	2	5	9	14	16	20

Left Right+Pivot

5 Compare, if a[left]<a[pivot]

	a[left]	<	a[pivot]		**Pivot=14**
Left=	10	<	14	TRUE	Left++
Left=	7	<	14	TRUE	Left++
Left=	2	<	14	TRUE	Left++
Left=	5	<	14	TRUE	Left++
Left=	9	<	14	TRUE	Left++
Left=	**14**				**Here, right=pivot=14. Terminate the loop and end the pass**

After Pass 2, Pivot=14 is at its best position

4	10	7	2	5	9	14	16	20	50	60	80

Sub-array-1 Sub-array 2 Sub-array 3

PASS 3: Consider the left Sub-array i.e. sub-array 1

1 Initialize Left =0, Right = Max-1 and as given Pivot = last element

4	10	7	2	5	9

Left+Pivot Right

2 Compare, if a[right]>a[pivot]

	a[right]	>	a[pivot]		**Pivot=4**
Right=	9	>	4	TRUE	Right--
Right=	5	>	4	TRUE	Right--
Right=	**2**	>	4	FALSE	Swap Right and Pivot and Set Pivot = Right

2	10	7	4	5	9

Left Right+Pivot

3 Compare, if a[left]<a[pivot]

	a[left]	<	a[pivot]		**Pivot=4**
Left=	2	<	4	TRUE	Left++
Left=	**10**	<	4	FALSE	Swap Left and Pivot and Set Pivot = Left

2	4	7	10	5	9

Left+Pivot Right

4 Compare, if a[right]>a[pivot]

	a[right]	>	a[pivot]		**Pivot=4**
Right=	10	>	4	TRUE	Right--
Right=	7	>	4	TRUE	Right--
Right=	**4**				**Here, right=pivot=4. Terminate the loop and end the pass**

After Pass 3, Pivot=4 is at its best position

2	4	7	10	5	9	14	16	20	50	60	80

Single Sub-array 1 Sub-array 2 Sub-array 3

PASS 4: Consider the left Sub-array i.e. sub-array 1

1 Initialize Left =0, Right = Max-1 and as given Pivot = last element

7	10	5	9

	Left+Pivot			Right

2 Compare, if a[right]>a[pivot]

	a[right]	>	a[pivot]	**Pivot=7**
Right=	9	>	7	TRUE Right--
Right=	**5**	>	7	FALSE Swap Right and Pivot and Set Pivot = Right

5	**10**	**7**	**9**
Left		Right+Pivot	

3 Compare, if a[left]<a[pivot]

	a[left]	<	a[pivot]	**Pivot=7**
Left=	5	<	7	TRUE Left++
Left=	**10**	<	7	FALSE Swap Left and Pivot and Set Pivot = Left

5	**7**	**10**	**9**
	Left+ Pivot	Right	

4 Compare, if a[right]>a[pivot]

	a[right]	>	a[pivot]	**Pivot=7**
Right=	10	>	7	TRUE Right--
Right=	**7**			**Here, right=pivot=7. Terminate the loop and end the pass**
				After Pass 4, Pivot=7 is at its best position

2	**4**	**5**	**10**	**7**	**9**	**14**	**16**	**20**	**50**	**60**	**80**
Single	Sub-array 1			Single	Sub-array 2				Sub-array 3		

PASS 5: Consider the left Sub-array i.e. sub-array 1

1 Initialize Left =0, Right = Max-1 and as given Pivot = last element

5	**10**
Left+ Pivot	Right

2 Compare, if a[right]>a[pivot]

	a[right]	>	a[pivot]	**Pivot=5**
Right=	10	>	5	TRUE Right--
Right=	**5**			**Here, right=pivot=5. Terminate the loop and end the pass**
				After Pass 5, Pivot=5 is at its best position

2	**4**	**5**	**10**	**7**	**9**	**14**	**16**	**20**	**50**	**60**	**80**
Single			Single	Single	Single		Sub-array 2			Sub-array 3	

PASS 6: Consider the sub-array 2

1 Initialize Left =0, Right = Max-1 and as given Pivot = last element

16	**20**
Left+ Pivot	Right

2 Compare, if a[right]>a[pivot]

	a[right]	>	a[pivot]	**Pivot=16**
Right=	20	>	16	TRUE Right--
Right=	**20**			**Here, right=pivot=16. Terminate the loop and end the pass**
				After Pass 6, Pivot=16 is at its best position

2	**4**	**5**	**10**	**7**	**9**	**14**	**16**	**20**	**50**	**60**	**80**
Single			Single	Single	Single			Single		Sub-array 3	

PASS 7: Consider the sub-array 3

1 Initialize Left =0, Right = Max-1 and as given Pivot = last element

60	**80**

	Left+ Pivot		Right											
2	Compare, if a[right]>a[pivot]													
	a[right]	>	a[pivot]		**Pivot=60**									
Right=	80	>	60	TRUE	Right--									
Right=	**60**				**Here, right=pivot=60. Terminate the loop and end the pass**									
					After Pass 7, Pivot=60 is at its best position									
	2	**4**	**5**	**10**	**7**	**9**	**14**	**16**	**20**	**50**	**60**	**80**		
	Single			Single		Single			Single			Single		

After Pass 7, the array is sorted as shown.