



AMRITA
VISHWA VIDYAPEETHAM

DESIGN AND ANALYSIS OF
ALGORITHMS
LAB WORKBOOK
WEEK – 6

NAME : ROHITH S

ROLL NUMBER : CH.SC.U4CSE24140

CLASS : CSE-B

Question 1: Write a program to perform Quick Sort by taking First Element, Last Element and a Random Element as Pivot Element for the given numbers:

157, 110, 147, 122, 149, 151, 111, 141, 112, 123, 133, 117

CODE:

```
C quicksort.c > ...
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4
5  void swap(int *a,int *b)
6  {
7      int t=*a;
8      *a=*b;
9      *b=t;
10 }
11
12 void printArray(int a[],int n)
13 {
14     for(int i=0;i<n;i++)
15         printf("%d ",a[i]);
16     printf("\n");
17 }
18
19 int partitionFirst(int a[],int low,int high)
20 {
21     int pivot=a[low];
22     int i=low+1;
23     int j=high;
24     while(i<=j)
25     {
26         while(i<=high&&a[i]<=pivot)
27             i++;
28         while(a[j]>pivot)
29             j--;
30         if(i<j)
31             swap(&a[i],&a[j]);
32     }
33     swap(&a[low],&a[j]);
34     return j;
35 }
```

```
37 void quickSortFirst(int a[],int low,int high)
38 {
39     if(low<high)
40     {
41         int p=partitionFirst(a,low,high);
42         quickSortFirst(a,low,p-1);
43         quickSortFirst(a,p+1,high);
44     }
45 }
46
47 int partitionLast(int a[],int low,int high)
48 {
49     int pivot=a[high];
50     int i=low-1;
51     for(int j=low;j<high;j++)
52     {
53         if(a[j]<=pivot)
54         {
55             i++;
56             swap(&a[i],&a[j]);
57         }
58     }
59     swap(&a[i+1],&a[high]);
60     return i+1;
61 }
62
63 void quickSortLast(int a[],int low,int high)
64 {
65     if(low<high)
66     {
67         int p=partitionLast(a,low,high);
68         quickSortLast(a,low,p-1);
69         quickSortLast(a,p+1,high);
70     }
71 }
```

```
73 int partitionRandom(int a[],int low,int high)
74 {
75     int r=low+rand()% (high-low+1);
76     swap(&a[r],&a[high]);
77     return partitionLast(a,low,high);
78 }
79
80 void quickSortRandom(int a[],int low,int high)
81 {
82     if(low<high)
83     {
84         int p=partitionRandom(a,low,high);
85         quickSortRandom(a,low,p-1);
86         quickSortRandom(a,p+1,high);
87     }
88 }
89
90 int main()
91 {
92     int a[]={157,110,147,122,149,151,111,141,112,123,133,117};
93     int n=sizeof(a)/sizeof(a[0]);
94     srand(time(0));
95     int a1[n],a2[n],a3[n];
96     for(int i=0;i<n;i++)
97     {
98         a1[i]=a[i];
99         a2[i]=a[i];
100        a3[i]=a[i];
101    }
102    quickSortFirst(a1,0,n-1);
103    quickSortLast(a2,0,n-1);
104    quickSortRandom(a3,0,n-1);
105    printArray(a1,n);
106    printArray(a2,n);
107    printArray(a3,n);
108    return 0;
109 }
```

OUTPUT:

```
110 111 112 117 122 123 133 141 147 149 151 157  
110 111 112 117 122 123 133 141 147 149 151 157  
110 111 112 117 122 123 133 141 147 149 151 157  
PS C:\Users\123sr\OneDrive\Desktop\COLLEGE STUFF\SEM IV\DAA LAB\Week 6> [ ]
```

WORKING:

Quick Sort
Working:-

i) First element as Pivot
Part 1

1) 162, 110, 142, 122, 111, 149, 151, 141, 123, 112, 117, 133
Pivot L → R → R → R → R → R → R → R → R → R → R
 $A[R] = 133 < 162$
 $R = 11$
 $L \geq R$
 $\therefore \text{Swap}$

Part 2)

133, 110, 147, 122, 111, 149, 151, 141, 123, 112, 117
 $L \rightarrow L$ $R \rightarrow R$
Pivot = 133
 $L = 1 \rightarrow 110$
 $R = 10 \rightarrow 117$
 $10 \leq 133 \rightarrow L++$
 $A[L] = 147 > 133$
 $L = 2$
Move R-
 $117 < 123$
Swap
133, 110, 117, 122, 111, 149, 151, 141, 123, 112, 147
 $L \rightarrow L$ $R \rightarrow R$
Swap(149, 112)
133, 110, 117, 122, 111, 112, 151, 141, 123, 149, 147
 $L \rightarrow L$ $R \rightarrow R$
Swap(151, 123)
133, 110, 117, 122, 111, 112, 123, 141, 151, 149, 147
 $R \rightarrow R$ $L \rightarrow L$
Swap with pivot
123, 110, 117, 122, 111, 112, 133, 141 & 151, 149, 147

Pass 3 $P_{pivot} = 123$
Left = $[125, 110, 117, 122, 111, 112]$
Right = $[141, 151, 149, 147]$

Arrangement
 $[123, 110, 117, 122, 111, 112]$
 $\begin{array}{cccccc} & & \downarrow & & & \\ L & \nearrow & R & \searrow & C & R \\ Swap(112, 113) \end{array}$
 $\rightarrow [112, 110, 117, 122, 111, 123]$

Pass 4 :- $P_{pivot} = 112$

$[112, 110, 117, 122, 111, 123]$
 $\begin{array}{cccccc} L & \downarrow & R & \downarrow & C & R \\ \nearrow & & \searrow & & \nearrow & \searrow \\ Swap(117, 111) \end{array}$
 $\rightarrow [112, 110, 111, 122, 117, 123]$
 $\begin{array}{cccccc} & \nearrow & & \nearrow & & \\ R & L & \nearrow & \searrow & & \\ Swap(111, 122) & (Pivot Swap) \end{array}$
 $[111, 110, 112, 122, 117]$

$L \rightarrow [111, 110] \rightarrow$ Sorted $\rightarrow [110, 111]$

$R \rightarrow [112, 117] \rightarrow$ Swap $\rightarrow [117, 122]$

Left side :-

$\Rightarrow [110, 111, 112, 117, 122, 123]$

Right Side Sorting :-

$[141, 151, 149, 147]$
 $P_{pivot} = 141$
 $\begin{array}{cccccc} & \nearrow & \nearrow & \nearrow & & \\ 141 & 151 & 149 & 147 & & \\ \downarrow & & & & & \end{array}$

Swap (Pivot with L)
 $[151, 141, 149, 147]$

$P_{pivot} = 151$
 $[151, 141, 149, 147]$
 $\begin{array}{cccccc} & & \downarrow & & & \\ & & R & & & \\ Swap(R, L) \end{array}$
 $[142, 141, 149, 151]$
 $P_{pivot} = 142$
 $[147, 141, 149, 151]$
 $\begin{array}{cccccc} & & \downarrow & & & \\ & & L & & & \\ Swap(L, R) \end{array}$
 $Swap(R, L)$
 $\rightarrow [141, 142, 149, 151]$

Merge two arrays

$[110, 111, 112, 117, 122, 123, 141, 142, 149,$
 $151]$

ii) Last element as Pivot

Pass L R
[117, 110, 149, 122, 111, 149, 151, 141, 123, 112, 147, 153]
Pivot = 153

Swap (L, R)
[117, 110, 149, 122, 111, 149, 151, 141, 123, 112, 147, 153]
L_L R_R

Swap (L, R)
[117, 110, 112, 122, 111, 149, 151, 141, 123, 112, 147, 153]
L_L R_R

Swap (L, R)
[117, 110, 112, 122, 111, 149, 151, 141, 149, 147, 157, 153]
L_L R_R

Swap (Pivot, L)

[117, 110, 112, 122, 111, 123, 153, 141, 149, 147, 152, 151]

Left Array :-

[117, 110, 112, 122, 111, 123]

Right Array :-

[141, 149, 147, 157, 151]

Pivot (Left Array):

[117, 110, 112, 122, 111, 123]
L_L R_R

Swap (111, 123)

[117, 110, 112, 122, 111, 123]
L_L R_R

Pivot:

[117, 110, 112, 122, 123, 113]
L_L R_R L_R

(i) Swap (L, R)

[110, 117, 112, 122, 123, 111]
L_L R_R

Swap (Pivot, R)

\Rightarrow [110, 111, 112, 122, 123, 113]

Sorted:-

Final array obtained :-

[110, 111, 112, 122, 123, 113, 141, 149, 147, 152, 153]

Right Array :-

[141, 149, 147, 151, 153]
L_L R_R L_R R_R

Swap (?, Pivot)

[141, 149, 147, 151, 153]
L_L R_R L_R R_R

Swap (L, R)

[141, 149, 147, 151, 153]
L_L R_R L_R R_R

Swap (Pivot, R)

[141, 147, 149] \rightarrow [151, 153]

L_R R_R

\Rightarrow [110, 111, 112, 122, 123, 113, 141, 149, 151, 153]

iii) Random element or pivot:-

A) $[153, 110, 143, 122, 111, 149, 151, 141, 123, 112, 117, 153]$
 $\downarrow_{L \rightarrow P}$ $\downarrow_{L \rightarrow R}$

B) $[153, 110, 143, 122, 111, 149, 151, 141, 123, 112, 117, 157]$
 $\downarrow_{L \rightarrow P}$ $\downarrow_{L \rightarrow R}$

Swap (L, R)

ii) $[153, 110, 143, 122, 111, 149, 112, 141, 123, 112, 117, 157]$

Left Array

$[153, 110, 143, 122, 111, 149, 112, 141, 123, 112]$

We take 149 as pivot

iii) $[153, 110, 143, 122, 111, 149, 117, 141, 123, 112]$
 $\downarrow_{L \rightarrow R}$ $\downarrow_{L \rightarrow P}$ $\downarrow_{L \rightarrow R}$

Swap (L, R)

iv) $[112, 110, 143, 122, 111, 149, 117, 141, 123, 133]$
 $\downarrow_{L \rightarrow L}$ $\downarrow_{L \rightarrow P}$ $\downarrow_{L \rightarrow P}$

Swap (L, R)

v) $[112, 110, 143, 122, 111, 149, 143, 141, 123, 133]$
 $\downarrow_{L \rightarrow L}$ $\downarrow_{L \rightarrow P}$

Swap (L, R)

vi) $[112, 110, 143, 122, 111, 149, 141, 143, 133]$

Swap (P, R)

vii) $[112, 110, 117, 122, 111, 123, 149, 141, 143, 133]$

Left Array :-

$[112, 110, 117, 122, 111]$

RB :-

$[149, 141, 143, 133]$

Solving LA:-

117 as pivot
 $[112, 110, 117, 122, 111]$
 $\downarrow_{L \rightarrow L}$ $\downarrow_{L \rightarrow P}$ $\downarrow_{L \rightarrow R}$

$[112, 110, 117, 122, 111]$
 \uparrow_R

$[112, 110, 117, 111, 122]$
 \downarrow_P \downarrow_R \uparrow_L

Swap (P, R)
 $\Rightarrow [112, 110, 122, 113, 122]$

Left Array :-

$[112, 110, 111]$

Leb-P = 111

$[112, 110, 111]$
 \downarrow_L \downarrow_R

\Rightarrow Swap (L, R)
 $\Rightarrow [110, 112, 111]$

Solving Right Array :-

$[149, 141, 117, 122, 111]$

Let $P = 122$

$\begin{matrix} 149, \\ 141, \\ 117, \\ 122, \\ \downarrow P \end{matrix}, \dots$

Swap (L, R)

$[111, 141, 117, 122, 149]$

$\begin{matrix} \uparrow \\ L \\ \uparrow \\ R \end{matrix}$

Swap (L, R)

$[111, 117, 141, 122, 149]$

$\begin{matrix} \uparrow \\ ? \\ \uparrow \\ L \\ \uparrow \\ R \end{matrix}$

Swap (L, P)

$[111, 117, 122, 141, 149]$

$L \leftarrow R$

$\Rightarrow [110, 111, 112, 122, 123, 133, 141, 147, 149, 151, 157]$

$\begin{matrix} \uparrow \\ 117 \end{matrix}$

ANALYSIS:

Time Complexity: $O(N^2)$: It takes $O(N^2)$ time in the worst case because each partition processes all elements while reducing the problem size by only one element at a time.

Space Complexity: $O(N)$: $O(N)$ space in the worst case because recursive calls can go as deep as the number of elements when partitions are unbalanced