

```

#include <bits/stdc++.h>
1  using namespace std;
2
3  //function declarations;
4  void Pivot_Position(int [],int );
5  void Sort(int[],int,int);
6  void Set(int[],int [],int);
7  void Swap_places(int *no1,int *no2);
8  void Display_array(int[],int );
9  int choose_pivot(int,int [],int,int);
10 int Middle(int [],int ,int );
11 int partition(int [],int ,int ,int,int &,int);
12
13 int main()
14 {
15     int n;
16     cout<<"Enter the number of elements to generate in the array :\t";
17     cin>>n;
18     int arr[n];
19     srand(time(0)); //seeding the srand method
20     for(int i=0;i<n;i++)
21         arr[i]=rand()%10;
22     cout<<"\n\nThe Array generate is : ";
23     for(int i=0;i<n;i++)
24         cout<<arr[i]<<" ";
25
26     Pivot_Position(arr,n);
27 }
28 int choose_pivot(int ch,int arr[],int l,int u)
29 {
30     int *a=&arr[l]; //lower
31     int *c=&arr[u]; //upper
32     int *b=&arr[(l+u)/2]; //middle
33     switch(ch)
34     {
35         case 1:return l;
36         case 2:return ((*a>=*c)?((*a<=*b)?a:((*b>=*c)?b:c)):((*c<=*b)?c:((*b>=*a)?b:a))-arr;
37         case 4:return (l+u)/2; //return middle
38         default : return -1;
39     }
40 }
41
42 int Middle(int arr[],int l,int u)
43 {
44     int sum=0;
45     for(int i=l;i<=u;i++)
46         sum+=arr[i];
47     return sum/(u-l+1);
48 }
49
50 int partition(int arr[],int l,int u,int n,int &comp,int ch)
51 {
52     int p;
53     int x;
54     int j=l-1;
55     if(ch!=3)
56     {
57         p=choose_pivot(ch,arr,l,u);
58         x=arr[p];
59         cout << "\n\n";
60     }
61     else
62     {
63         p=-1;
64         x=Middle(arr,l,u);
65

```

```

66     cout << "\n";
67 }
68 for(int i=l;i<=u;i++)
69 {
70     if(i!=p)
71     {
72         comp++;
73         if(arr[i]<=x)
74         {
75             j++;
76             if(j==p)p=i;
77             Swap_places(&arr[i],&arr[j]);
78         }
79     }
80 }
81 if(ch!=3)Swap_places(&arr[j+1],&arr[p]);
82 Display_array(arr,n);
83 return(ch==3&&j==u)?j:(j+1);
84 }
85
86 void Sort(int arr[],int n,int ch)
87 {
88     stack<int> Up;
89     stack<int> Low;
90     Up.push(n-1);
91     Low.push(0);
92     int comp=0;
93     while(!Up.empty()&&!Low.empty())
94     {
95         int u=Up.top();
96         int l=Low.top();
97         Up.pop();
98         Low.pop();
99         if(u>l)
100         {
101             int p=partition(arr,l,u,n,comp,ch);
102
103             Up.push(p-1);
104             Low.push(l);
105             Up.push(u);
106             (ch!=3)?Low.push(p+1):Low.push(p);
107         }
108     }
109     cout<<"\nTotal Comparisons :t"<<comp;
110 }
111
112 void Set(int arr[],int Array[],int n)
113 {
114     for(int i=0;i<n;i++)
115         arr[i]=Array[i];
116 }
117
118 void Pivot_Position(int arr[],int n)
119 {
120     int Array[n];
121
122     for(int i=0;i<n;i++)
123         Array[i]=arr[i];
124
125     cout<<"\n\nPivot as the first Element :- \n";
126     Sort(arr,n,1);
127
128     Set(arr,Array,n);
129     cout<<"\n\nPivot at Median:- \n";
130     Sort(arr,n,2);
131
132

```

```
133 Set(arr,Array,n);
134 cout<<"\n\nPivot as mean:- \n";
135 Sort(arr,n,3);
136
137 Set(arr,Array,n);
138 cout<<"\n\nPivot at the middle element :- \n";
139 Sort(arr,n,4);
140
141 }
142
143 void Swap_places(int *no1,int *no2)
144 {
145 //call by reference and Swap_places the places
146
147 int temp=*no1;
148 *no1=*no2;
149 *no2=temp;
150
151 }
152
153 void Display_array(int arr[],int n)
154 {
155 for(int i=0;i<n;i++)
156 cout<<arr[i]<<" ";
157 }
```