Date:__________

Title Page

Name: ASHFAQ AHMAD

Reg No: 19PWCSE1795

Section: B

Paper: Mid-term

Subject: DSA.

Submitted to:
Dr- Nasru minullah

Date: 28-4-2021

—— xx —— xx —— xxx

Date:

Answer No = 01

linear Search would perform best as Compare to Binary Search due to following reasons.

(1) the list is unsorted and is only to be search once.

(2) The list will need Sorting following the Search operation (due to say an Insertion) Since the restoring will dominate the time Complexity of overall task

(3) the data Structure is not random access.

(4) This is no knowledge of Data that Could aid Searching.

—— xx —— xx —— xx —— xx

P T O

Answer No 3:

My Res No = 19pwcsE 1798

= 1 9 1 2 3 4 5 1 7 9 8

= 1 9 2 3 4 5 7

```
# Include <iostream>
Using name-space std;
int main()
{
    int arr [] = {1 9, 2, 3, 4, 5}
  for ( int i=0 ; i<5 ; i++)
    { for ( j = i+1 ; j<6 ; i++ )
     { if (arr [j ]<= arr[i])
        { int x = arr [j];
          arr[j ] = arr [i]
    }     arr[i] = x ;
   }
  }
```

—— xx —— xx

Answer No : 04

Name = Ashfaq

~~Sorting~~

Sorted array

A f h q S.

first mid last

First = 1

last = 5

mid = 3

last (mid] h

Key : A

~~Ashfa~~

A f h q S

1st mid last

First = 2

last = 5

mid = 4

las [mid] = q

Key = A

—— XX —— xx —— xx —— xx

Answer No : 05

```
#include <iostream>
using namespace std;
int main()
{
   int arr[]={1,9,7,5}
  for (int i=0; i<=2; i++)
  {
    for ( j=i+1; j<4; j++)
   {  if (arr[j] < arr[i])
     { int  x = arr[j];
        arr [j] = arr[i];
         arr [i] = x;
       }
      }
    }
for (int i=0; i<4; i++)
 {
    cout << arr[i];
    cout << endl;
  }
  return  0;
 }
```

OUTPUT:

```
1
5
7
9
```

output

Note!

In case of quick sort first we select one element as pivot

[1, 9, 7, 5]
pivot

[1, 9, 7, 5]
↑ pivot
↑ right
left

we will move left pointer to right

we will swap 5, 7,

[1, 9, 5, 7]

Now swap (5, 9)

[1, 5, 9, 7]

Now swap (9, 7)

[1, 5, 7, 9]

——XX ——XX —— XX X.

Answer No 6:

my Res No: 19PWCSE1795

```
* include <iostream>
using namespace std;
int main()
{
   int arr[] = {1,9,7,5}
for (int i=0; i<=2; i++)
 {
  for (j=i+1; j<4; j++)
  { if (arr[j]< arr[i])
    {
     int x=arr[j];
     arr[j]= arr[i];
     arr[i]= x;
    }
   }
 }
for (int i=0; i<4; i++)
 { cout << arr[i];
   cout << endl;
 }
 return 0;
};
```

xx —— xx ——→ xx

the END

Answer No:07!

Bubble Sort!

Reg No: 19PWCSE1795

1	9	7	5	91	12	98
↑	↑					

no Swap

1	9	7	5	91	12	98
	↑	↑				

Swap

1	7	9	5	91	12	98
		↑	↑			

swap

1	7	5	9	91	12	98
			↑	↑		

no Swap

1	7	5	9	91	12	98
				↑	↑	

swap

1	7	5	9	12	91	98
	↑	↑				

→ In first Itteration highest no reached to its position

Swap

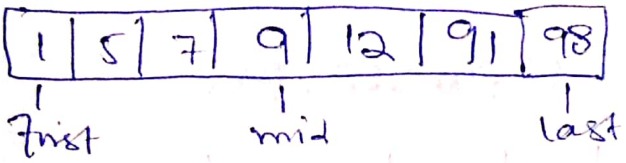
1	5	7	9	12	91	98
	↑	↑				

no swap

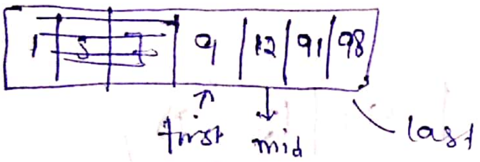
1	5	7	9	12	91	98

This is the sorted arry no further swaping possible

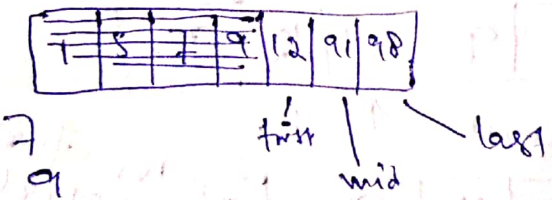
Binary search:



Key = 91
first = 1
last = 9
mid = 5
list [mid] = 9



first = 3
last = 9
mid = 7
list [mid] = 12
Key = 91



first = 7
last = 9
mid = ~~33~~ 46
list [mid] = 91 key found
key = 91

———— xx ———— xx ———— xx