VISHWAKARMA INSTITUTE OF TECHNOLOGY
DATA STRUCTURE ASSIGNMENT

NAME	Arpit Sudhir Vidhale
ROLL NO.	60
DIVISION	CS-D
BATCH	B3
PRN NO.	12111229

ASSIGNMENT 1

Code :

```
#include <stdio.h>
int Max (int a[], int n);
void Bucket_Sort (int a[], int n);
void swap (int *a, int *b);
int partition (int array[], int low, int high);
void quickSort (int array[], int low, int high);
void printArray (int arr[], int N);
struct Faculty
{
  char name[30];
  int Id;
  char subject_Code[5];
  char class_Name[5];
};
int main ()
{

  int n = 5;
  int ch;
  int data[n];
  struct Faculty faculties[n];
 for (int i = 0; i < n; i++)
```

```
 {
   printf ("Faculty %d:- \n", i + 1);
   printf ("Name: ");
   scanf ("%s", faculties[i].name);
   printf ("Id: ");
   scanf ("%d", &faculties[i].Id);
   printf ("subject Code: ");
   scanf ("%s", faculties[i].subject_Code);
   printf ("class Name: ");
   scanf ("%s", faculties[i].class_Name);
   data[i] = faculties[i].Id;
   printf ("\n");
 }
printf ("Enter your choice: \n1.Bucket Sort\n2.Quick
Sort\n\n"); scanf ("%d", &ch);
switch (ch)
 {
  case 1:
   {
            Bucket_Sort (data, n);
            printArray (data, n);
         printf ("******* All faculties Details ********\n");

         for (int i = 0; i < n; i++)
          {
            if (data[i] == faculties[0].Id)
             {
                  printf ("\t%s\t%d\t%s\t%s\n",
                           faculties[0].name, faculties[0].Id,
                           faculties[0].subject_Code,
                           faculties[0].class_Name);
             }
            else if (data[i] == faculties[1].Id)
             {
```

```
                printf ("\t%s\t%d\t%s\t%s\n",
                        faculties[1].name, faculties[1].Id,
                        faculties[1].subject_Code,
                        faculties[1].class_Name);
            }
           else if (data[i] == faculties[2].Id)
            {
                printf ("\t%s\t%d\t%s\t%s\n",
                        faculties[2].name, faculties[2].Id,
                        faculties[2].subject_Code,
                        faculties[2].class_Name);
            }
           else if (data[i] == faculties[3].Id)
            {
                printf ("\t%s\t%d\t%s\t%s\n",
                        faculties[3].name, faculties[3].Id,
                        faculties[3].subject_Code,
                        faculties[3].class_Name);
            }
           else if (data[i] == faculties[4].Id)
            {
                printf ("\t%s\t%d\t%s\t%s\n",
                        faculties[4].name, faculties[4].Id,
                        faculties[4].subject_Code,
                        faculties[4].class_Name);
            }
         }
       break;
 }
case 2:
 {
           quickSort (data, 0, n - 1);
           printArray (data, n);
```

```
    printf ("******* All faculties Details
********\n"); for (int i = 0; i < n; i++)
 {
   if (data[i] == faculties[0].Id)
    {
        printf                    ("\t%s\t%d\t%s\t%s\n",
                faculties[0].name,          faculties[0].Id,
                faculties[0].subject_Code,
                faculties[0].class_Name);
    }
   else if (data[i] == faculties[1].Id)
    {
        printf                    ("\t%s\t%d\t%s\t%s\n",
                faculties[1].name,          faculties[1].Id,
                faculties[1].subject_Code,
                faculties[1].class_Name);
    }
   else if (data[i] == faculties[2].Id)
    {
        printf                    ("\t%s\t%d\t%s\t%s\n",
                faculties[2].name,          faculties[2].Id,
                faculties[2].subject_Code,
                faculties[2].class_Name);
    }
   else if (data[i] == faculties[3].Id)
    {
        printf                    ("\t%s\t%d\t%s\t%s\n",
                faculties[3].name,          faculties[3].Id,
                faculties[3].subject_Code,
                faculties[3].class_Name);
    }
   else if (data[i] == faculties[4].Id)
    {
```

```
                printf ("\t%s\t%d\t%s\t%s\n",
                        faculties[4].name, faculties[4].Id,
                        faculties[4].subject_Code,
                        faculties[4].class_Name);
            }
          }
        break;
     }
    default:
     {
        printf ("Invalid
        Choice!!"); break;
     }
   }
}
//functions
int Max (int a[], int n)
{
  int max = a[0];
  for (int i = 1; i < n; i++)
    if (a[i] > max)
      max = a[i];
  return max;
}
void Bucket_Sort (int a[], int n)
{
  int max = Max (a, n);
  int bucket[max], i;
  for (int i = 0; i <= max; i++)
    {
      bucket[i] = 0;
    }
  for (int i = 0; i < n; i++)
```

```
    {
      bucket[a[i]]++;
    }
  for (int i = 0, j = 0; i <= max; i++)
    {
      while (bucket[i] > 0)
          {
            a[j++] = i;
            bucket[i]--;
          }
    }
}
void swap (int *a, int *b)
{
  int t = *a;
  *a = *b;
  *b = t;
}
int partition (int array[], int low, int high)
{
  int pivot = array[high];
  int i = (low - 1);
  for (int j = low; j < high; j++)
    {
      if (array[j] <= pivot)
          {
            i++;
            swap (&array[i], &array[j]);
          }
    }
  swap (&array[i + 1],
  &array[high]); return (i + 1);
}
```

```
void quickSort (int array[], int low, int high)
{
  if (low < high)
    {
      int pi = partition (array, low, high);
      quickSort (array, low, pi - 1);
      quickSort (array, pi + 1, high);
    }
}
void printArray (int arr[], int N)
{
  for (int i = 0; i < N; i++)
     printf ("%d ", arr[i]);
  printf ("\n");
}
```

Output :

```
Faculty 1:-
Name: ram
Id: 60
subject Code: ds
class Name: cs
Faculty 2:-
Name: sham
Id: 55
subject Code: it
class Name: csd
Faculty 3:-
Name: arun
Id: 65
subject Code: eee
class Name: cdb
Faculty 4:-
Name: arpit
Id: 58
subject Code: ieee
class Name: csa
Faculty 5:-
Name: sudhir
Id: 62
subject Code: re
class Name: csc
```

```
Enter your choice:
1.Bucket Sort
2.Quick Sort

1
55 58 60 62 65
******* All faculties Details ********
    sham    55  it  csd
    arpit   58  ieee    csa
    ram 60  ds  cs
    sudhir  62  re  csc
    arun    65  eee cdb
```