**Practical-5**

* Implement of counting sort

**Code:**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int scount = 0;

int getMax(int a[], int n)

{

    int max = a[0];

scount++;

    for (int i = 1; i< n; i++, scount++)

    {

        if (a[i] > max)

        {

            max = a[i];

scount++;

        }

    }

    return max;

scount++;

}

void countSort(int a[], int n)

{

    int output[n + 1];

scount++;

    int max = getMax(a, n);

scount++;

    int count[max + 1];

scount++;

    for (int i = 0; i<= max; ++i,scount++)

    {

        count[i] = 0;

scount++;

    }

    for (int i = 0; i< n; i++,scount++)

    {

        count[a[i]]++;

scount++;

    }

    for (int i = 1; i<= max; i++,scount++)

    {

        count[i] += count[i - 1];

scount++;

    }

    for (int i = n - 1; i>= 0; i--,scount++)

    {

        output[count[a[i]] - 1] = a[i];

scount++;

        count[a[i]]--;

scount++;

    }

    for (int i = 0; i< n; i++,scount++)

    {

        a[i] = output[i];

scount++;

    }

}

int main()

{

int sz;

 clock\_t start, end;

double time\_taken;

time\_t t;

printf("Enter the size of array: ");

scanf("%d", &sz);

    int randArray[sz], i;

srand((unsigned)time(&t));

    for (i = 0; i<sz; i++)

    {

randArray[i] = rand() % 100;

    }

printf("\nElements of the array: ");

    for (i = 0; i<sz; i++)

    {

printf("%d ", randArray[i]);

    }

    start = clock();

countSort(randArray, sz);

printf("\nAfter sorting array elements are :");

    for (i = 0; i<sz; i++)

    {

printf("%d ", randArray[i]);

    }

    end = clock();

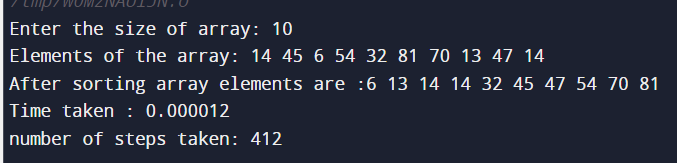
time\_taken = ((double)(end - start)) / CLOCKS\_PER\_SEC;

printf("\nTime taken : %f", time\_taken);

printf("\nnumber of steps taken: %d ",scount);

}

**Output:**

****

|  |  |  |
| --- | --- | --- |
| **Values** | **Steps** | **Time Taken** |
| **5** | **386** | **0.012** |
| **50** | **792** | **0.11** |
| **500** | **4404** | **7.1** |
| **5000** | **40404** | **8.63** |
| **50000** | **400401** | **64.94** |

0

50000

100000

150000

200000

250000

300000

350000

400000

450000

5

50

500

5000

5000

step

step

0

1

2

3

4

5

6

7

8

9

10

5

50

500

5000

50000

time

time