**Objective:**

|  |  |  |
| --- | --- | --- |
| **Compiled Date** | **Scheduled Date** | **Submitted Date** |
| **22-09-20** | **22-09-20** | **22-09-20** |

Algorithm of Counting Sort

Counting-Sort(A,B,K)

1. for i <- 0 to k
2. do C[i] <- 0
3. for j <- 1 to length[A]
4. do C[A[j]] <- C[A[j]]+1
5. ……C[i] now contains the number of element equal to i.
6. for i <- 1 to k
7. do C[i] <- C[i]+C[i-1]
8. ……..C[i] now contains the number of elements less than or equal to i.
9. for j <- length[A] down to 1
10. do B[C[A[j]]] <- A[j]
11. C[A[j]] <- C[A[j]] – 1

Program of Counting Sort

#include<stdio.h>

int count=0;

int getdata(int[],int);

void putdata(int[],int);

int main()

{

void counting\_sort(int[],int[],int,int);

int a[100],b[100],n,k;

printf("Enter the number of elements:");

scanf("%d",&n);

k=getdata(a,n);

count++;

counting\_sort(a,b,k,n);

count++;

printf("Elements Entered:\n");

putdata(a,n);

printf("Sorted element:\n");

putdata(b,n);

printf("Count for %d is:%d",n,count);

return 0;

}

int getdata(int a[],int n)

{

int i,k;

printf("Enter the %d Element for sorting\n",n);

for(i=1;i<=n;i++)

{

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

scanf("%d",&a[i]);

if(k<a[i])

k=a[i];

}

return k;

}

void putdata(int b[], int n)

{

int i;

for(i=1;i<=n;i++)

{

printf("%d\t",b[i]);

}

printf("\n");

}

void counting\_sort(int a[],int b[],int k,int n)

{

int i,j,C[100];

count++;

for(i=0;i<=k;i++)

{

count++;

C[i]=0;

count++;

}

count++;

for(j=1;j<=n;j++)

{

count++;

C[a[j]]=C[a[j]]+1;

count++;

}

count++;

for(i=1;i<=k;i++)

{

count++;

C[i]=(C[i]+C[i-1]);

count++;

}

count++;

for(j=n;j>=1;j--)

{

count++;

b[C[a[j]]]=a[j];

count++;

C[a[j]]=C[a[j]]-1;

count++;

}

count++;

}

Counting Sort Graph

|  |  |  |  |
| --- | --- | --- | --- |
| Input Size | Best Case | Avarage Case | Worst Case |
| 5 | 54 | 54 | 54 |
| 10 | 96 | 99 | 99 |
| 15 | 144 | 144 | 144 |
| 20 | 189 | 189 | 189 |
| 25 | 234 | 234 | 234 |