Counting sort

#include<iostream>

using namespace std;

void counting\_sort(int \*arr,int n)

{

//largest in the array assuming no>0

int largest=-1;

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

{

largest=max(largest,arr[i]);

}

//create a frequency array

int \*freq=new int[largest+1]{0};

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

{

freq[arr[i]]++;

}

//put the element back into the array a by the reading frequency

int j=0;

for(int i=0;i<=largest;i++)

{

while(freq[i]>0)

{

arr[j]=i;

freq[i]--;

j++;

}

}

}

int main()

{

int arr[]={1,8,94,32,56,74,85,14,2,3};

int n=sizeof(arr)/sizeof(int);

counting\_sort(arr,n);

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

{

cout<<arr[i]<<" ";

}

return 0;

}

Bucket Sort

#include<iostream>

#include<vector>

#include<algorithm>

using namespace std;

class Student{

public:

int marks;

string name;

};

void bucket\_sort(Student s[],int n)

{

//assuming marks are in range 1-100

vector<Student>v[101];

//O(N) time

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

{

int m=s[i].marks;

v[m].push\_back(s[i]);

}

//Iterate over the vector and print the Student

for(int i=100;i>=0;i--)

{

for(vector<Student>::iterator it=v[i].begin();it!=v[i].end();it++)

{

cout<<(\*it).marks<<" "<<(\*it).name<<endl;

}

}

}

//Bucket sort to sort an array of Student

int main()

{

Student s[100];

int n;

cin>>n;

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

{

cin>>s[i].marks>>s[i].name;

}

bucket\_sort(s,n);

return 0;

}

DNF SORT

#include<iostream>

using namespace std;

void dnfSort(int a[],int n)

{

int lo=0;

int hi=n-1;

int mid=0;

while(mid<=hi)

{

if(a[mid]==0)

{

swap(a[mid],a[lo]);

lo++;

mid++;

}

if(a[mid]==1)

{

mid++;

}

if(a[mid]==2)

{

swap(a[mid],a[hi]);

hi--;

}

}

return;

}

int main()

{

int a[]={2,1,1,0,0,1,0,1,2,1,0,2};

int n=sizeof(a)/sizeof(int);

dnfSort(a,n);

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

{

cout<<a[i]<<" ";

}

}

Wave Sort

#include<iostream>

using namespace std;

int main()

{

int a[]={1,3,4,2,7,8};

//sort the array in wave form

int n=sizeof(a)/sizeof(int);

for(int i=0;i<n;i+=2)

{

//Prev element

if(i!=0 and a[i]<a[i-1])

{

swap(a[i],a[i-1]);

}

//next Element

if(i!=(n-1) and a[i]<a[i+1])

{

swap(a[i],a[i+1]);

}

}

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

{

cout<<a[i]<<" ";

}

}