1.Bubble Sort

```
bubbleSort(a){
for i=0 to n{
if(a[i]>a[i+1]{
swap(a[i],a[i+1])
}
}
```

2.Selection Sort

```
selectionSort(){

for(i=0;i<n-1;i++){
min=i;
for(j=i+1;j<n;j++){
if(a[j]<a[min]){
min=j;}
}
if(i!=min){
swap(a[i],a[min])}
}
}
```

3.Quick Sort

```
quickSort(l,h)
{
if(l<h){
j=partition(l,h);
quickSort(l,j);
quickSort(j+1,h);
```

```
}

partition(l,h)
{
pivot=a[l]
i=l,j=h;
while(i<j){
do{
i++;
} while(a[i]<=pivot);

do{
j--;
} while(a[j]>pivot);

  if(i<j)
        swap(a[i],a[j])
}
swap(a[l],a[j]);
returnj;
}
```

4.Merge Sort

```
mergeSort(l,h){

if(l<h){
mid=l+h/2;
mergeSort(l,mid);
mergeSort(mid+1,h);
merge(l,mid,h);}
}



merge(l,mid,h){
i=l;j=mid+1;k=l;
while(i<=mid && j<=h)
{
if(a[i]<=a[j]){
b[k]=a[i];
i++;}
else{
b[k]=a[j]
j++;}
k++;}
if(i>mid){
```

```
while(j<=l){
b[k]=a[j];
j++;k++;}
}
else{
while(i<=mid){
b[k]=a[i];
i++;k++;}
}
}
```