**冒泡排序**：

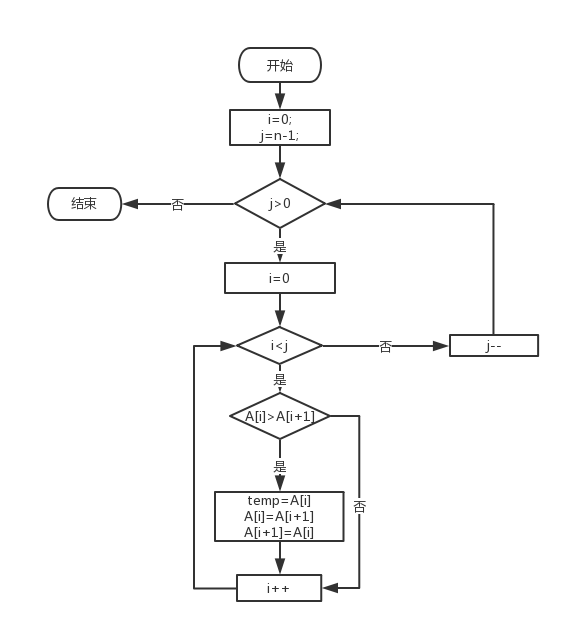
假设总共有n个数字需要排序

初始是i=0;j=n-1;

1. 让第i个数字和第i+1个数字，如果第i个数字比第i+1个数字大，就交换它们的位置

2．i=i+1,如果i>=j,到第3步。否则回到第1步

3.j=j-1,如果j<=0,算法结束，否则回到第1步



**代码**：

#include<iostream>

using namespace std;

void mSort(int \*a,int n);

int main(){

int a[10]={9,8,7,6,5,4,3,2,1,0};

mSort(a,10);

for(int i=0;i<10;i++){

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

}

cout<<endl;

}

void mSort(int \*a,int n){

int temp=0;

for(int j=n-1;j>0;j--){

for(int i=0;i<j;i++){

if(a[i]>a[i+1]){

temp=a[i];

a[i]=a[i+1];

a[i+1]=temp;

}

}

}

}

**最坏情况**：O(n^2)

**平均情况**：O(n^2)

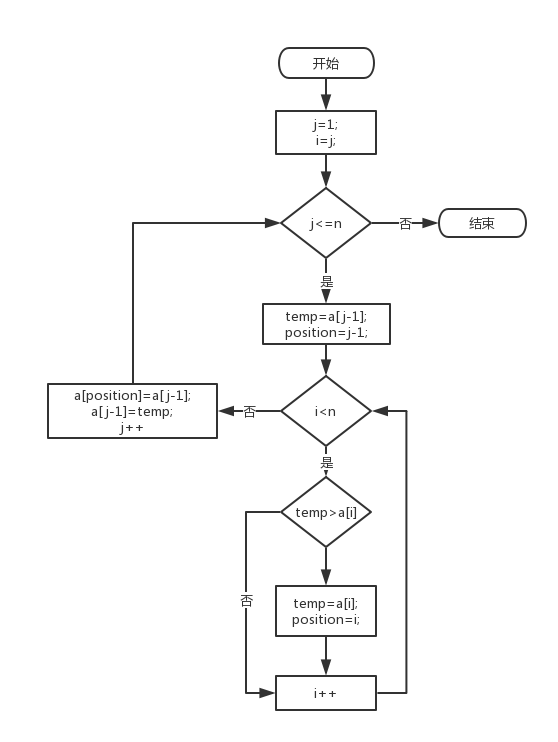
**最好情况**：O(n)

**选择排序**

假设总共有n个需要排序的元素

1. i=j;j=1;;
2. temp=a[j-1] ,position=i-1;
3. 如果temp>a[i],则temp=a[i],position=i;
4. i++，如果i>n,到第6步。否则到第5步。
5. a[position]=a[j-1] a[j-1]=temp;
6. j++，如果j>n,算法结束，否则到第1步

**流程图**



**代码**

#include<iostream>

using namespace std;

void mySort(int \*a,int n);

int main(){

int a[10]={9,8,7,6,5,4,3,2,1,0};

mySort(a,10);

for(int i=0;i<10;i++){

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

}

cout<<endl;

}

void mySort(int \*a,int n){

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

int temp=a[j-1];

int position=j-1;

for(int i=j;i<n;i++){

if(temp>a[i]){

temp=a[i];

position=i;

}

}

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

a[j-1]=temp;

}

}

**最坏情况**：O(n^2)

**平均情况**：O(n^2)

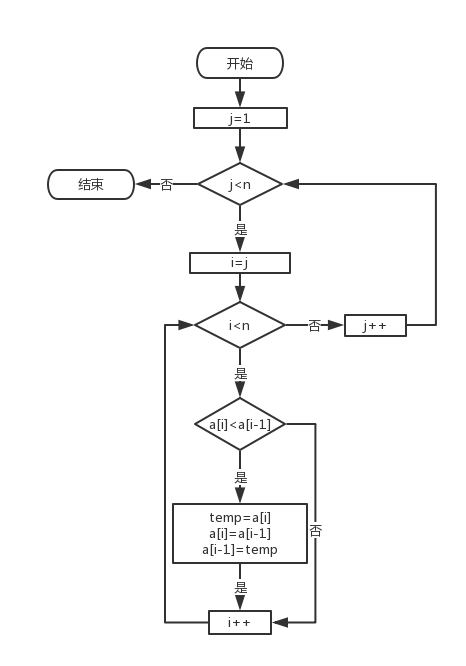
**最好情况**：O(n^2)

**插入排序**

假设有n个元素需要排序

1. j=1
2. i=j
3. 如果a[i]>a[i+1],交换它们位置
4. i--,如果i>0,直接到第3步
5. j++,如果j==n,算法结束，否则到第2步

**流程图**



**代码**

#include<iostream>

using namespace std;

void mySort(int \*a,int n);

int main(){

int a[10]={9,8,7,6,5,4,3,2,1,0};

mySort(a,10);

for(int i=0;i<10;i++){

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

}

cout<<endl;

}

void mySort(int \*a,int n){

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

int i=j;

for(;i>0;i--){

if(a[i-1]>a[i]){

int temp=a[i-1];

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

a[i]=temp;

}

}

}

}