折半插入

|  |
| --- |
| #include<stdio.h>  #include<string.h>  int main(){  int bin\_insertsort(int data[],int n);  void print(int data[],int n);  int data[8]={1,5,7,3,0,8,2,6};  bin\_insertsort(data,8);  print(data,8);  return 0;    }  int bin\_insertsort(int k[],int n){  int i,j,low,high,middle;  int temp;  for(i=1;i<n;i++){  temp = k[i];  low=0;  high=i-1;  while(low<=high){  middle=(low+high)/2;  if(temp<k[middle]) high=middle-1;  else low = middle+1;  }  for(j=i-1;j>=low;j--){  k[j+1]=k[j];  }  k[low]=temp;  }  }  void print(int k[],int n){  //output the  printf("%d\n",k[n]);  for(int i=0;i<n;i++){  printf("%d\n",k[i]);  }  } |

改成指针形式

|  |
| --- |
| #include<stdio.h>  #include<string.h>  int main(){  int bin\_insertsort(int data[],int n);  void print(int data[],int n);  int data[8]={1,5,7,3,0,8,2,6};  int \*p;  p=data;  bin\_insertsort(p,8);  print(data,8);  return 0;    }  int bin\_insertsort(int \*arr,int n){  int i,j,low,high,middle;  int temp;  for(i=1;i<n;i++){  temp = \*(arr+i);  low=0;  high=i-1;  while(low<=high){  middle=(low+high)/2;  if(temp<\*(arr+middle)) high=middle-1;  else low = middle+1;  }  for(j=i-1;j>=low;j--){  \*(arr+j+1)=\*(arr+j);  }  \*(arr+low)=temp;  }  }  void print(int \*arr,int n){  //output the  printf("%d\n",\*(arr+n));  for(int i=0;i<n;i++){  printf("%d\n",\*(arr+i));  }  } |

谢尔排序

|  |
| --- |
| //  // main.c  // demo  //  // Created by wby on 2019/11/5.  // Copyright © 2019 wby. All rights reserved.  //  #include<stdio.h>  int main(){  void shellsort(int data[],int n);  void print(int data[],int n);  int data[8]={1,5,7,3,0,8,2,6};  shellsort(data,8);  print(data,8);  return 0;  }  void shellsort(int k[],int n){  int i,j,flag,gap=n;  int temp;  while(gap>1){  gap=gap/2;  do{  flag=0;  for(i=0;i<n-gap;i++){  j=i+gap;  if (k[i]>k[j]){  temp=k[i];  k[i]=k[j];  k[j]=temp;  flag=1;  }  }  }while(flag !=0);  }    }  void print(int k[],int n){  //output the  printf("%d\n",k[n]);  for(int i=0;i<n;i++){  printf("%d\n",k[i]);  }  } |