《计算机程序设计》作业 **№-06及第5次上机**

# 作业四内容要点： 数组应用、调试方法

【姓名 马宇骁 学号 PB19151769 】

**【要求】**

1. **在计算机上编程程序，加上必要的注释。**
2. **上机实验，经助教检查通过后，复制源码并记录实验结果，完成报告。**
3. **实验报告：记录调试及改错过程；****知识点或方法技巧的收获心得.**

**1 、找零问题**

某人购买物品一共花了x元（x<=100），用100元现金去支付，售货员需要找零，请设计程序给出一个找零方案，并使找零的张数最少。设现有人民币面值包括：100元、50元、20元、10元、5元、1元、5角和1角。

编程要求：

输入购买物品的费用x，并判断其合理性（ 0.0 ~ 100.0）

对输入合理的费用，给出找零的方案（即每种钱币的数量），使之找零的钱币张数最少

当x包含1角以下的金额时，按照四舍五入到角之后再进行找零。

程序运行示例：

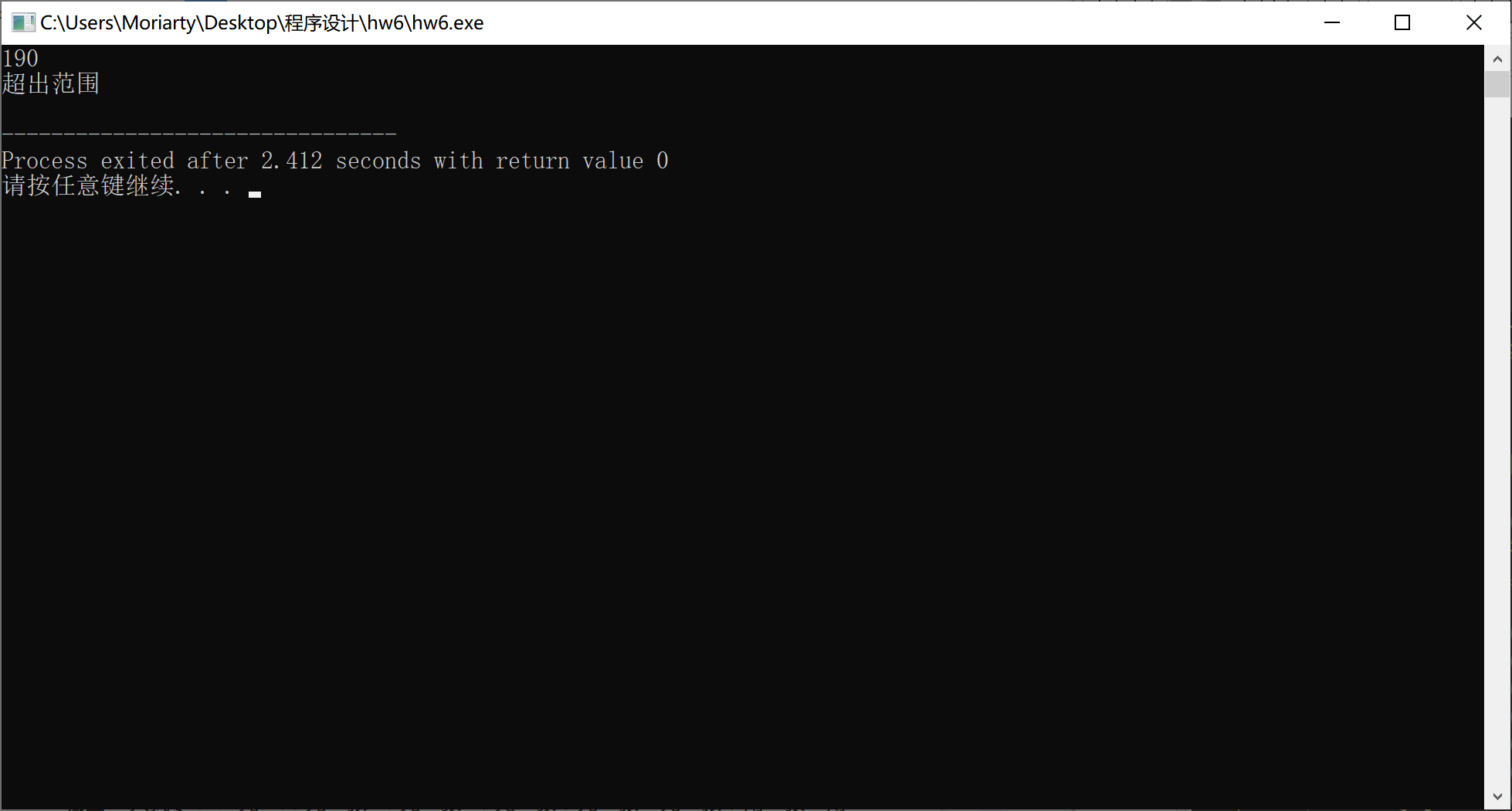
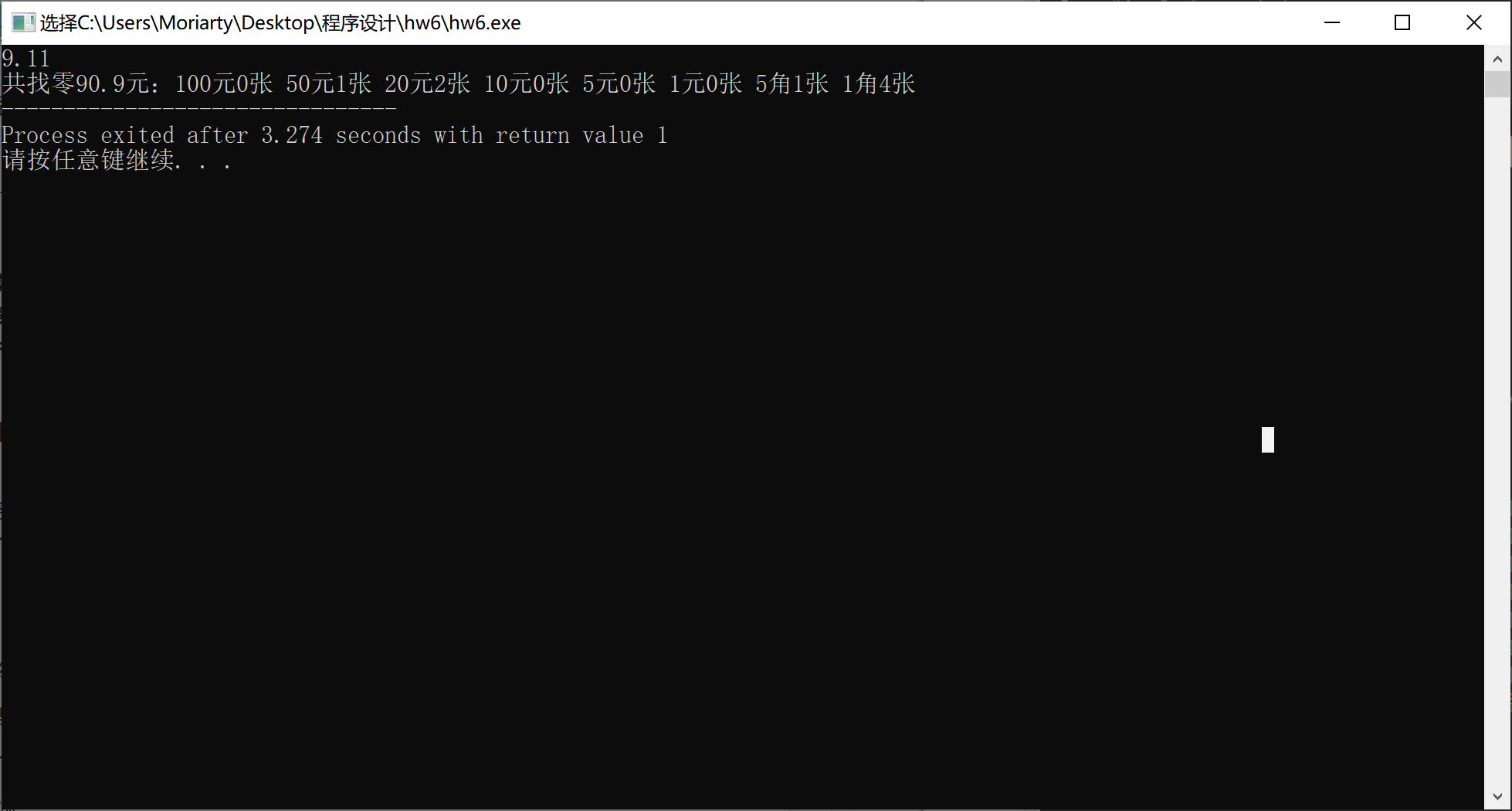
**输入（单位 元）**：12.34

**输出**： 共找零87.7元：50元1张 20元1张 10元1张 5元1张 1元2张 5角1张 1角2张

1. 【源码】

* #include<stdio.h>
* #include<math.h>
* int main(){
* float x;
* scanf("%f",&x);
* //判断
* if(x<0 || x>100){
* printf("超出范围\n");
* return 0;
* }
* else{
* //一位小数点四舍五入
* float x2;
* x2 = 100 - ((int)(10\*x + 0.5))/10.0;
* printf("共找零%.1f元：",x2);
* //计算找零钱，张数最少
* int cnt[8] = {0};
* cnt[0] = int(x2/100);
* cnt[1] = int((x2-cnt[0]\*100)/50);
* cnt[2] = int((x2-cnt[0]\*100-cnt[1]\*50)/20);
* cnt[3] = int((x2-cnt[0]\*100-cnt[1]\*50-cnt[2]\*20)/10);
* cnt[4] = int((x2-cnt[0]\*100-cnt[1]\*50-cnt[2]\*20-cnt[3]\*10)/5);
* cnt[5] = int((x2-cnt[0]\*100-cnt[1]\*50-cnt[2]\*20-cnt[3]\*10-cnt[4]\*5)/1);
* cnt[6] = int((x2-cnt[0]\*100-cnt[1]\*50-cnt[2]\*20-cnt[3]\*10-cnt[4]\*5-cnt[5])/0.5);
* cnt[7] = int((x2-cnt[0]\*100-cnt[1]\*50-cnt[2]\*20-cnt[3]\*10-cnt[4]\*5-cnt[5]-cnt[6]\*0.5)/0.1);
* printf("100元%d张 50元%d张 20元%d张 10元%d张 5元%d张 1元%d张 5角%d张 1角%d张",cnt[0],cnt[1],cnt[2],cnt[3],cnt[4],cnt[5],cnt[6],cnt[7]);
* return 1;
* }
* }

1. 【运行结果】

* 
* 

1. 【实验报告】

* 在实现结束之后注意到其实可以使用俩个数组进行循环操作减少代码量。

**2 、顺序查找和二分查找的比较**

现有500个随机数 [0 .. 1999] 存放于文本文件rand500.txt中。请编程序, 利用输入重定向读入这500个随机数；另设定一个数组 int key[10] ={  *10个整型常量，取值范围[0, .. , 1999]，取值尽量随机分布* }; 然后完成以下任务：

1). 用**顺序查找**的方法，在500个随机数中分别查找key[0] .. key[9] , 并记录查找过程中进行了多少次比较。分别计算并输出：当key值存在和不存在时，查找一个key值平均进行了多少次比较？

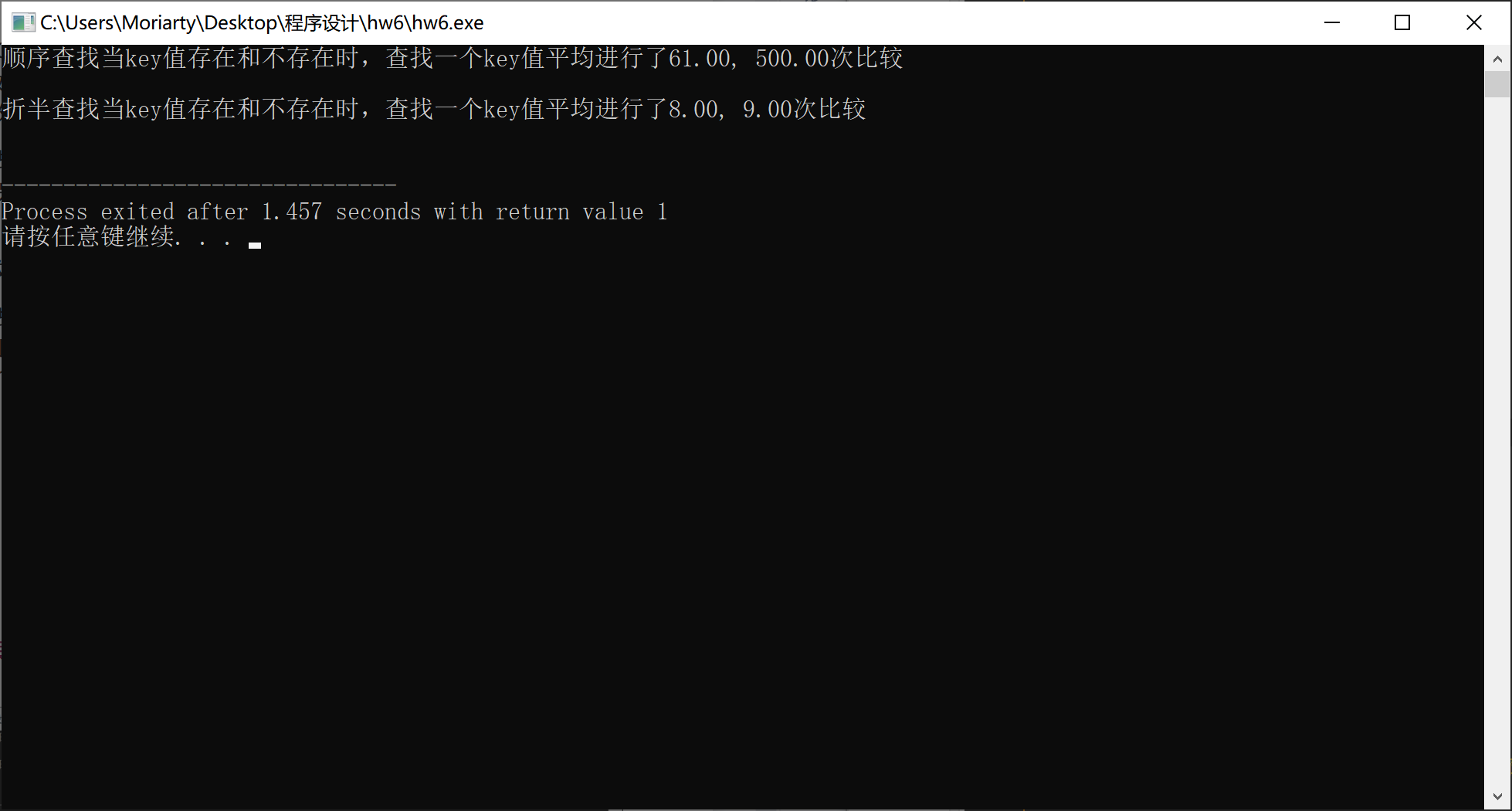
2）. 对500个随机数进行排序（方法任选一种：冒泡法、选择法、插入法）

3). 用**二分查找**的方法： 在排序后的500个随机数中分别查找key[0] .. key[9] , 并记录查找过程中进行了多少次比较。分别计算并输出：当key值存在和不存在时，查找一个key值平均进行了多少次比较？

1. 【源码】

* int main(){
* //读取
* freopen("random.txt","r",stdin);
* int r[500];
* for(int i=0;i<500;i++){
* scanf("%d",&r[i]);
* }
* fclose(stdin);
* //生成随机数
* int key[10];
* srand((unsigned)time(NULL));
* for(int i = 0; i < 10; ++i){
* key[i] = (int)(round(1.0 \* rand() / RAND\_MAX \* 2000));
* }
* //1)顺序查找
* int cmp[10]={0};
* float cmp1=0,cmp2=0;
* int cnt=0,j;
* for(int i=0; i<10; i++){
* for(j = 0; j<500; j++){
* cmp[i] +=1;
* if(key[i] == r[j]){
* break;
* }
* }
* if(j==500){
* cmp2 = cmp2 + cmp[i];
* cnt += 1;
* }
* else{
* cmp1 = cmp1 + cmp[i];
* }
* }
* cmp2 = cmp2/cnt;
* cmp1 = cmp1/(10-cnt);
* printf("顺序查找当key值存在和不存在时，查找一个key值平均进行了%.2f, %.2f次比较\n\n",cmp1,cmp2);
* //2)冒泡排序 (从小到大)
* int temp;
* for(int i=0;i<500-1;i++){
* for(j=0;j<500-i-1;j++){
* if(r[j]>r[j+1]){
* temp = r[j];
* r[j]=r[j+1];
* r[j+1]=temp;
* }
* }
* }
* //3)二分查找
* int cmp0[10]={0};
* cmp1=0;
* cmp2=0;
* cnt=0;
* int low, high, mid;
* for(int i=0; i<10; i++){
* low=0;
* high=500-1;
* mid = (low+high)/2;
* while(high != low){
* cmp0[i] += 1;
* if(key[i] == r[mid]){
* break;
* }
* else if(key[i] < r[mid]){
* high = mid;
* }
* else{
* low = mid+1;
* }
* mid = (low+high)/2;
* }
* if (high == low){ //没找到
* cmp2 = cmp2 + cmp0[i];
* cnt += 1;
* }
* else{
* cmp1 = cmp1 + cmp0[i];
* }
* }
* cmp2 = cmp2/cnt;
* cmp1 = cmp1/(10-cnt);
* printf("折半查找当key值存在和不存在时，查找一个key值平均进行了%.2f, %.2f次比较\n\n",cmp1,cmp2);
* getchar();
* return 1;
* }

1. 【运行结果】

* 

1. 【实验报告】

* 考验学生的查找相关的知识，十分有趣。