姓名：钟代琪 学号：2018302130155

代码：

#include<stdio.h>

#include<stdlib.h>

#include<time.h>

//获得较大数

int max(int a, int b) {

if (a > b) return a;

else return b;

}

int main()

{

//创建output.txt文件

FILE\* fp = nullptr;

fopen\_s(&fp, "output.txt", "wt");

srand((unsigned)time(nullptr));

int arr[8];

for (int i = 0; i < 8; i++) {

arr[i] = rand() \* 20 / RAND\_MAX - 10;

//判断是否重复

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

if (arr[j] == arr[i]) arr[i] = rand() \* 20 / RAND\_MAX - 10;

}

fprintf\_s(fp, "%d ", arr[i]);

}

fclose(fp);

int opt[8];

opt[0] = arr[0];//边界

int begin[8] = { 0 }, end = 0;//将每个子状态opt的序列开头保存在begin里面

for (int i = 1; i < 8; i++) {

int temp = opt[i - 1];

opt[i] = max(arr[i], temp + arr[i]);//状态转移方程

//判断opt[i]的开头是多少

if (arr[i] > temp + arr[i]) begin[i] = i;

else begin[i] = begin[i - 1];

}

int max = opt[0];

for (int i = 1; i < 8; i++) {

if (opt[i] > max) {

max = opt[i]; end = i;//判断子状态的最优解，并得出序列结尾

}

}

int start = begin[end];

fopen\_s(&fp, "output.txt", "at");

fprintf\_s(fp,"\n最大和连续子序列是" );

for (; start <= end; start++) {

fprintf\_s(fp,"%d ", arr[start]);

}

fprintf(fp, "\n最大和是%d", max);

fclose(fp);

return 0;

}

调试心得：

1.第一次尝试使用动态规划的思想解决问题，中间多次出现了逻辑错误，经过多次修正后调试成功。但是还是需要更多练习和讲解来提高。

2.知道了如何设置随机数，rand()和srand()的使用方法。利用循环判断设置不重复的随机数。

3.输入子序列时利用了数组来保存每个子状态的开头项数，感觉空间复杂度较高，但没想出更好的方法。