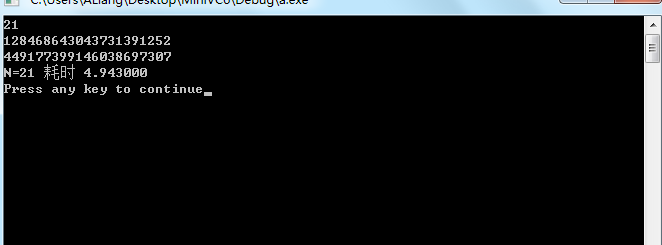
水仙花数

21位时用时4.94秒



程序：

#include<stdio.h>

#include<string.h>

#include<time.h>

#define MAX 30

int N=21,top=0;//设置位数N

char kk[50][MAX];//存储输出结果

void add(int dest[],int src1[],int src2[]){

int flag=0,n , i=1 ;

if(src1[0]>src2[0]){

dest[0] = src1[0] ;

n = src2[0]+1 ;

memcpy(dest+n,src1+n,(MAX-n)\*4);

}

else{

dest[0] = src2[0] ;

n = src1[0] + 1 ;

memcpy(dest+n,src2+n,(MAX-n)\*4);

}

for(; i < n ; i++ ){

dest[i] = src1[i] + src2[i] + flag ;

if(dest[i]>9){

dest[i] -= 10 ;

flag = 1;

}

else

flag=0;

}

if(flag){

do{

dest[i]=dest[i]+1;

if(dest[i] > 9 )

dest[i] -= 10 ;

else

break;

i++;

}while(true);

if(i>dest[0])

dest[0]=i;

}

}

void squre(int sum[],int m,int n){//求m的n次方，并将结果放到sum中

int f=0;

if(n==1){

sum[0]=1;

sum[1]=m;

return;

}

squre(sum,m,n-1);

for(int i = 1 ; i <= sum[0] ; i++ ){

sum[i] = sum[i]\*m + f ;

f = sum[i]/10 ;

sum[i] %= 10 ;

}

if(f){

sum[0]++ ;

sum[i] = f;

}

}

void push(int sum[]){

for(int i=1;i<=N;i++)

kk[top][N-i]=sum[i]+'0';//将结果存放到字符串中

kk[top][N]='\0';

top++;

}

void popPrint(){//从小到大打印输出

int i,j,small;

for(i=0;i<top;i++){

small=i;

for(j=i+1;j<top;j++){

if(strcmp(kk[small],kk[j])>0)

small=j;

}

printf("%s\n",kk[small]);

strcpy(kk[small],kk[i]);

}

}

//判断是否满足组合条件，

void equal(int sum[],int dest[]){

int a[10]={0},i=0;

for(i=1;i<=N;i++){

a[sum[i]]++;

a[dest[i]]--;

}

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

if(a[i])

return;

push(sum);

}

void main(){

clock\_t t1 ;

clock\_t t2 ;

int b[10][MAX]={0},i;

int mem[MAX]={0};

int Csum[MAX][MAX] ={0} ;

int \*sum ;

int sNext = 1 , k;

scanf("%d",&N);

t1 = clock();

b[1][0]=1;

b[1][1]=1;

for(i=2;i<10;i++)

squre(b[i],i,N);//计算出0-9的n次方

for(i=1;i<=N ; i++)

mem[i] = 9 ;

Csum[0][0]=1;

while(mem[1]){

sum = Csum[sNext] ;

i=sNext-1;

do{

i++;

add(Csum[i],Csum[i-1],b[mem[i]]);

}while(i<N && Csum[i][0]<=N );

sum = Csum[i];

if(sum[0]==N){

equal(sum,mem);

}

else

if(sum[0]>N){//修饰大的部分

k = --mem[i] ;

sNext = i ;

for(;i<=N ; i++)

mem[i]=k;

continue;

}

else{//修饰小的部分

for(i=N-1 ; i && (mem[i]-mem[N])<2 ; i-- );

if(i){

k = --mem[i];

sNext = i;

for(i++; i<=N ; i++)

mem[i] = k ;

}

else{

break;

}

continue;

}

if(mem[N]){

mem[N]--;

sNext = N ;

}

else{

for( i=N-1 ; mem[i]==0 ; i--);

k=--mem[i];

sNext = i ;

for(i++ ; i <= N ; i++)

mem[i] = k;

}

}

popPrint();

t2 = clock();

printf("N=%d 耗时 %f\n",N,(t2-t1)\*1.0/1000);

}