实验环境：

|  |  |
| --- | --- |
| 处理器 | Intel Core i5-4210H@2.9GHz~3.5GHz |
| 内存 | 8GB 1600MHz DDR3L |
| 显卡 | NVIDIA Geforce GTX960M+Intel HD4600 |
| 硬盘 | 500GB 5400rpm+128G SSD |
| 显示器 | 15.6’’ 1920x1080 |
| 操作系统 | Windows 10 Home |
| 编译环境 | Codeblocks 13.12+MingW+GCC 4.8.1 |

第一题两种算法，一个用大根堆每次取出最大值并维护，时间复杂度O(klogn)，k为次数；

第二种是线性查找，复杂度O(kn)。

第二题高斯消元没啥说的。

断点调试就不截图了……用的编译器又不一样。我保证会用……

代码：

1.

#include<iostream>

#include<time.h>

#include<stdlib.h>

using namespace std;

int a[200],n=199,m;

void swapa(int x,int y)

{

int t=a[x];

a[x]=a[y];

a[y]=t;

}

void sift\_up(int x)

{

while(x)

{

if (a[x]>a[x>>1]) swapa(x,x>>1);

x=x>>1;

}

}

void sift\_down(int x)

{

int t=x;

if (((x<<1)+1<=n)&&(a[(x<<1)+1]>a[t])) t=(x<<1)+1;

if (((x<<1)<=n)&&(a[x<<1]>a[t])) t=x<<1;

if (t==x) return;

swapa(t,x);

sift\_down(t);

}

int main()

{

srand(time(NULL));

for (int i=1;i<n;i++)//用大根堆找出最大值

{

a[i]=rand();

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

}

cout<<endl;

for (int i=100;i<=n;i++) sift\_up(i);

for (int k=0;k<10;k++)

{

cout<<a[1]<<endl;

swapa(1,n);

n--;

sift\_down(1);

}

n=199;

a[0]=0;

for (int k=0;k<10;k++)//线性查找找出最大值

{

m=0;

for (int i=1;i<=n;i++) if (a[i]>a[m]) m=i;

cout<<a[m]<<endl;

a[m]=0;

}

return 0;

}

2.

#include<iostream>

#include<cmath>

#define E 0.00001

using namespace std;

float a[4][5]={1.1161,0.1254,0.1397,0.1490,1.5471,

0.1582,1.1675,0.1768,0.1871,1.6471,

0.2368,0.2471,0.2568,1.2671,1.8471,

0.1968,0.2071,1.2168,0.2271,1.7471};

int i,i2,j,k,n;

void printa()

{

for (int i = 0; i <n; ++i)

{

for(int j=0;j<=n;j++)cout<<a[i][j]<<" ";

cout<<endl;

}

}

void swapxy(int x,int y)

{

int k;

float temp;

for (k = 0; k <= n; ++k)

{

temp=a[x][k];

a[x][k]=a[y][k];

a[y][k]=temp;

}

}

void del(int l1,int l2,int code)

{

float p=a[l2][l1]/a[l1][l1];

if(fabs(a[l2][l1])<E) return;

if (code==0)

for (int i = l1; i < n; ++i)

{

a[l2][i]-=a[l1][i]\*p;//delete

}

else

for (int i = l1; i >= 0; --i)

{

a[l2][i]-=a[l1][i]\*p;//delete

}

a[l2][n]-=a[l1][n]\*p;//process the last element

}

int main()

{

n=4;

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

{

if (fabs(a[i][i])<E)

{

for (i2 = i+1; i2 < n; ++i2)

{

if(fabs(a[i2][i])>E)

{

swapxy(i,i2);

i2=n+1;

}

}

};

for (i2 = i+1; i2 < n; ++i2)

{

del(i,i2,0);//将第i2行化为0,消成上三角

}

}

for(i=n-1;i>=0;--i)

{

if(fabs(a[i][i])<E)

{

if (fabs(a[i][n])>E) {cout<<"No solution";return 0;} else {cout<<"Infinite solutions";return 0;}

}else

for (i2 = i-1; i2 >= 0; --i2)

{

del(i,i2,1);//将第i2行化为0,消成对角线

}

}

//printa();

for(i=0;i<n;i++)//print solution

{

cout<<"x("<<i+1<<")="<<a[i][n]/a[i][i]<<endl;

}

return 0;

}