扩展 kmp

```
#include <iostream>
#include <string>

using namespace std;

/* 求解 T 中 next[]，注释参考 GetExtend() */
void GetNext(string & T, int & m, int next[])
{
	int a = 0, p = 0;
	next[0] = m;

	for (int i = 1; i < m; i++)
	{
		if (i >= p || i + next[i - a] >= p)
		{
			if (i >= p)
				p = i;

			while (p < m && T[p] == T[p - i])
				p++;

			next[i] = p - i;
			a = i;
		}
		else
			next[i] = next[i - a];
	}
}

/* 求解 extend[] */
void GetExtend(string & S, int & n, string & T, int & m, int extend[], int next[])
{
	int a = 0, p = 0;
	GetNext(T, m, next);

	for (int i = 0; i < n; i++)
	{
		if (i >= p || i + next[i - a] >= p) // i >= p 的作用：举个典型例子，S 和 T 无一字符相同
		{
			if (i >= p)
				p = i;
```

```
            while (p < n && p - i < m && S[p] == T[p - i])
                p++;

            extend[i] = p - i;
            a = i;
        }
        else
            extend[i] = next[i - a];
    }
}

int main()
{
    int next[100];
    int extend[100];
    string S, T;
    int n, m;

    while (cin >> S >> T)
    {
        n = S.size();
        m = T.size();
        GetExtend(S, n, T, m, extend, next);

        // 打印 next
        cout << "next:     ";
        for (int i = 0; i < m; i++)
            cout << next[i] << " ";

        // 打印 extend
        cout << "\nextend: ";
        for (int i = 0; i < n; i++)
            cout << extend[i] << " ";

        cout << endl << endl;
    }
    return 0;
}
```

极角排序

```
/* ***********************************************
Author              :xdlove
Created Time    :2015 年 11 月 13 日 星期五 23 时 24 分 49 秒
File Name         :xdlove/codeforces/Educational_Codeforces_Round_1/C/C.cpp
 ********************************************** */
```

```
#pragma comment(linker, "/STACK:1024000000,1024000000")
#include <stdio.h>
#include <string.h>
#include <iostream>
#include <algorithm>
#include <memory.h>
#include <vector>
#include <queue>
#include <set>
#include <map>
#include <string>
#include <math.h>
#include <stdlib.h>
#include <time.h>

using namespace std;


#define REP_ab(i,a,b) for(int i = a; i <= b; i++)
#define REP(i, n) for(int i = 0; i < n; i++)
#define REP_1(i,n) for(int i = 1; i <= n; i++)
#define DEP(i,n) for(int i = n - 1; i >= 0; i--)
#define DEP_N(i,n) for(int i = n; i >= 1; i--)
#define CPY(A,B) memcpy(A,B,sizeof(B))
#define MEM(A) memset(A,0,sizeof(A))
#define MEM_1(A) memset(A,-1,sizeof(A))
#define MEM_INF(A) memset(A,0x3f,sizeof(A))
#define MEM_INFLL(A) memset(A,0x3f3f,sizeof(A))
#define mid (((l + r) >> 1))
#define lson l, mid, u << 1
#define rson mid + 1, r, u << 1 | 1
#define ls (u << 1)
#define rs (u << 1 | 1)


typedef long long ll;
typedef unsigned long long ull;
const int INF = 0x3f3f3f3f;
const ll INFLL = 0x3f3f3f3f3f3f3f3f;
const int MAXN = 1e5 + 5;
const int MAXM = MAXN;
const int mod = 1e9 + 7;
const long double pi = acos(-1.0);
```

```
const long double eps = 1e-8;

struct Point
{
	long double k;
	int id;
	bool operator < (const Point &a) const
	{
		return k < a.k;
	}
}p[MAXN];

int main()
{
	//freopen("in.txt","r",stdin);
	//freopen("out.txt","w",stdout);
	int n;
	cin>>n;
	REP(i,n)
	{
		int x,y;
		scanf("%d %d",&x,&y);
		p[i].k = atan2(y,x);
		p[i].id = i + 1;
	}
	sort(p,p + n);
	long double ans;
	if(p[0].k < 0) ans = p[0].k + 2 * pi - p[n - 1].k;
	else ans = p[n - 1].k - p[0].k;
	int a1 = p[n - 1].id,a2 = p[0].id;
	for(int i = 1; i < n; i++)
	{
		long double tp = p[i].k - p[i - 1].k;
		if(tp    < ans)
		{
			ans = tp;
			a1 = p[i].id;
			a2 = p[i - 1].id;
		}
	}
	cout<<a1<<" "<<a2<<endl;
	return 0;
}
```