

Đàñĩ đảảảểảí ềả Êĩ ø ề

Í ảí ửí ả÷ảí ềả	$C(x / a, b)$
Í ảềảñử ửí ả÷ảí ềề	$-\infty < x < +\infty$
Ỉ ảðảì ảòðử	Ỉ ảðảì ảòð ðảñĩ í ềí ảảí ềỷ a, í ảảềảí ả, í ảðảì ảòð ì ảñø ðảảả b > 0.
Ỉ ềí ấ í ñử (ồ óí ềөөё ảảđỉ ỷấ í ñề)	$bp^{-1}[(x-a)^2 + b^2]^{-1}$
Ì ảòảì ảò÷ảñềí ả í ảềảảí ềả	Í ả ñử ảñòảóảò
Ảềñĩ ảðñềỷ	Í ả ñử ảñòảóảò
Ồóí ềөөё ðảñĩ đảảảểảí ềỷ	$0.5 + \{arctg[(x-a)/b]\}/p$

Ñảỷử ñ ảðốảềì ề ðảñĩ đảảảểảí ềỷì ề

Í óí 1 ø ảí ềả ảảó í ảçảảềñềì ử õ í ảềí ảềí ảí ðảñĩ đảảảểảí ử õ í ảðì ảềủí ử õ ñềó÷ảềí ử õ ảảề÷ềí y_1, y_2 ñ ñảảảí ềì $\mu=0$ ề ñảả ảảðỏí ử ì í ðềềí í ảí ềảì $\sigma=1$ í í ả÷ềí ỷảòny ðảñĩ đảảảểảí ềỷ Êĩ ø ề ñ í ảðảì ảòðảì ề a=0, b=1.

Ñỏí ì ả ñ í ảçảảềñềì ử õ ñềó÷ảềí ử õ ảảề÷ềí $x_1 \sim C(a_1, b_1)$, í í ả÷ềí ỷảòny ðảñĩ đảảảểảí ềỷ Êĩ ø ề ñ í ảðảì ảòðảì ề, ðảảí ử ì ề ñỏí ì ả ñ í ảðảòñòảóỷ ử ề õ í ảðảì ảòðỉ ả ñềảảảảì ử õ:

$$x_1 + x_2 + \dots + x_n \sim C(a_1 + a_2 + \dots + a_n, b_1 + b_2 + \dots + b_n).$$

Ảảí ảðảөөё ñềó÷ảềí ử õ ÷ềñảề

Ềñĩ í ềüçỏỷ ñảả ảảðỏí ử ề ì ảỏí ả í ảðảỏỉ ảả í ð ðảảí í ì ảðỉ í ảí ðảñĩ đảảảểảí ềỷ, í í ềó÷ảảì óí ðì óềó: $r_C \sim tg[\pi(r-0.5)]$, ảảả r - ðảảí í ì ảðỉ í ðảñĩ đảảảểảí í í ả ñềó÷ảềí í ả ÷ềñềí, ả r_C í í ả÷ềí ỷảòny ðảñĩ đảảảểảí ềỷ Êĩ ø ề.

Ảử ÷ềñềảí ềả ô óí ềөөё ðảñĩ đảảảểảí ềỷ ề ảả ềảảí ềềểảề

Í ả í ðảảñòảảềỷảò í ềềảềềø ððỏảí í ñỏảề: ềñĩ í ềüçỏỷ ðny ềềø ử ô óí ềөөё, ảỏí ảỷử ềả ả ñỏả ảảðỏí ử ả ảềảềềí ðảềề (ềảề ả Ñề), ềềảí ả ñảì ỷử ề (ềảề ả Í ảñềảềả).

Ảềỷ ñ ảñảì ềảí ềảử õ ảñả-ðảềề í ðềảí ảó ềí ảử. Í í í í ề ðả ảñả ảả, ÷ỏí ềñĩ í ềüçỉ ảả í ử ả ñỏả ảảðỏ- í ử ả ô óí ềөөё ðảảí ðảỷ ð ñ í ảðảì ề÷ảí í í ề ðí ÷í ñòủỷ !

Ôàëë cauchy.h

```
#ifndef __CAUCHY_H__          /* To prevent redefinition*/
#define ENTRY    extern
#define LOCAL    static

ENTRY double cauchyDF(double x, double a, double b);
ENTRY double inv_cauchyDF(double q, double a, double b);

#define __CAUCHY_H__          /* Prevents redefinition */
#endif                        /* Ends #ifndef__CAUCHY_H__ */
```

Ôàëë cauchy.cpp

```
#include <assert.h>
#include <math.h>

const double pi=3.1415926;
const double half=0.5;

double cauchyDF(double x, double a, double b)
{
    assert(b > 0);
    return half+atan((x-a)/b)/pi;
}

double inv_cauchyDF(double q, double a, double b)
{
    assert((b > 0) && (q > 0) && (q < 1));
    return a+b*tan(pi*(q-half));
}

#ifdef TEST

#include <iostream.h>

void main(void)
{
    double a, b;

    while (1) {
        cout << "\n\n\rEnter a: ";
        cin >> a;
        if (a <= 0)
            break;
        do {
            cout << "Enter b: ";
            cin >> b;
        } while (b <= 0);

        for(double x=-8; x < 8; x += 0.1)
        {
            double y=cauchyDF(x, a, b);
            cout << "x=" << x << "\tcauchy=" << y;
            double z=inv_cauchyDF(y, a, b);
            cout << "\ti_cauchy=" << z << endl;
        }
    }
}
```

```
}  
#endif
```