

```
// sample8_2_1.cpp : メイン プロジェクト ファイルです。
```

```
#include "stdafx.h"
#include <iostream>
#include <iomanip>
```

```
using namespace System;
using namespace std;
```

```
//
// Wichmann/Hill generator...周期 = 約1.0e12
//
```

```
class WHrn {
public:
```

```
    unsigned __int32 x;
    unsigned __int32 y;
    unsigned __int32 z;
    double u;
```

```
    WHrn(): x(11111), y(22222), z(33333) {}
    virtual ~WHrn(){}
```

```
    WHrn(unsigned __int32 ix, unsigned __int32 iy, unsigned __int32 iz):
        x(ix), y(iy), z(iz) {}
```

```
    double uni(); // 0 < uni() < 1
```

```
};
//
// 正規乱数
//
```

```
class NormalRN: public WHrn {
public:
```

```
    NormalRN(){}
    virtual ~NormalRN(){}
```

```
    NormalRN(unsigned __int32 ix, unsigned __int32 iy, unsigned __int32 iz): WHrn(ix, iy, iz) {}
```

```
    double sqr( double x ){
        return x * x;
```

```
    }
    //
    // 標準正規乱数
    //
```

```
    double NormalRN::normal();
```

```
    //
    // 平均 m、標準偏差 s の正規乱数
    //
```

```
    double NormalRN::normalMS(double m, double s);
```

```
    //
    // 独立な標準正規乱数 n1 と n2 の対
    //
```

```
    void NormalRN::normalPair( double & n1, double & n2 );
```

```
};
```

```
double WHrn::uni(){
```

```
    do {
        x = (171 * x) % 30269;
        y = (172 * y) % 30307;
        z = (170 * z) % 30323;
        u = (x/30269.0) + (y/30307.0) + (z/30323.0);
        u = u - Math::Truncate(u);
    } while (u <= 0.0);
    return u;
```

```
};
//
// A Rejection Polar Method for Normal Variates
//
```

```
double NormalRN::normal(){
```

```
    double v1, v2, w;
    do {
```

```
    v1 = 2.0 * uni() - 1.0;
    v2 = 2.0 * uni() - 1.0;
    w = sqr(v1) + sqr(v2);
} while ((w >= 1.0) || (w <= 0.0));

double c = Math::Sqrt(-2.0 * Math::Log(w)/w);

return c * v1;
}

double NormalRN::normalMS(double m, double s){
    double v = s * normal() + m;
    return v;
}
//
// A Rejection Polar Method for Normal Variates
//
void NormalRN::normalPair( double & n1, double & n2 ){
    double v1, v2, w, c;
    do {
        v1 = 2.0 * uni() - 1.0;
        v2 = 2.0 * uni() - 1.0;
        w = sqr(v1) + sqr(v2);
    } while ((w >= 1.0) || (w <= 0.0));

    c = Math::Sqrt(-2.0 * Math::Log(w) / w);

    n1 = c * v1;
    n2 = c * v2;
}

int main(array<System::String ^> ^args)
{
    WHrn rn;
    NormalRN rn1, rn2(1, 2, 3);
    cout << "rn: x = " << rn.x << "   ¥ty = " << rn.y << "   ¥tz = " << rn.z << endl;
    cout << "rn1:x = " << rn1.x << "   ¥ty = " << rn1.y << "   ¥tz = " << rn1.z << endl;
    cout << "rn2:x = " << rn2.x << "   ¥ty = " << rn2.y << "   ¥tz = " << rn2.z << endl;
    for (int i = 0; i < 20; i++){
        cout << setw(7) << rn.uni() << "   ¥t" << rn1.normal();
        double n1, n2;
        rn2.normalPair(n1, n2);
        cout << "   ¥t" << n1 << "   ¥t" << n2 << endl;
    }

    Console::WriteLine();
    Console::WriteLine("Enterキーを押して終了。");
    Console::ReadLine();
    return 0;
}
```