```cpp
#include <iostream>
#include <cmath>
#include <random>

#include <cppqt.h>

using namespace std;


// zeichnet eine Linie von p1 nach p2 (Algorithmus ist verbesserungsfähig)
void zeichneLinie(Drawing& pic, IPoint2D p1, IPoint2D p2, DrawColour c)
{
    // bestimme Abstand von p1 und p2 und male entsprechend viele Punkte
    // dazwischen
    int len = round(norm(p2 - p1));
    IPoint2D r = (p2-p1);
    IPoint2D q;
    double delta;
    for(int k = 0; k <= len; ++k)
    {
        delta = static_cast<double>(k) / len;
        q = round(static_cast<DPoint2D>(p1) + delta * static_cast<DPoint2D>(r) );
        pic.drawPoint(q, c, false);
    }
}


int maindraw()
{
    Drawing pic(400, 400);
```

```
    pic.show();
    pic.setZoom(2);

    IPoint2D p1, p2;

    // Just some random numbers
    default_random_engine generator;
    uniform_int_distribution<int> distribution(0, 255);
    auto dice = bind(distribution, generator);


    while(true)
    {
        bind(distribution, generator);
        DrawColour c(dice(), dice(), dice());
        cout << "Eingabe von p1, p2: ";
        cin >> p1 >> p2;

        if (p1.x < 0 || p1.y < 0 || p2.x < 0 || p2.y < 0)
        break;

        zeichneLinie(pic, p1, p2, c);
    }

    cout << endl;
    IOThread::waitForWindow(60); //to close

    return 0;
}
```