

```
// Kontrollstrukturen.cpp : Definiert den Einstiegspunkt für die Konsolenanwendung.
//

#include "stdafx.h"
#include <iostream>
#include "Math.h"

using namespace std;

void ascii(){
    cout << "\nAnton als ASCII-Strom" << endl;
    string anton = "||||||\n|      |\n| o o |\n|  L  |\n|  ~  |";
    for(int i=0;i < anton.length();i++){
        cout << anton[i];
    }
    cout << endl;
    //Versuche auch
    for(int i=0;i < anton.length();i++){
        cout << (int)anton[i]; // Typ int erzwingen
    }
    cout << endl;
}

void kopfstand(){
    cout << "\nAnton's Kopfstand" << endl;
    string anton = "||||||\n|      |\n| o o |\n|  L  |\n|  ~  |";
    for(int i=anton.length()-1;i>=0;i--){
        cout << anton[i];
    }
}

void asciifolge(){
    cout << "Zahlenfolge decodieren" << endl;
    int a1=65, a2=114, a3=109, a4=101, a5=114, a6=32, a7=65, a8=110, a9=116, a10=111, a11=110;
    cout << (char)a1 << (char)a2 << (char)a3 << (char)a4 << (char)a5 << (char)a6 << (char)a7 << (char)a8 << (char)a9 << (char)a10 << (char)a11 << endl;
    cout << (char)a8 << (char)a9 << (char)a10 << (char)a11 << endl;
}

void reis(){
    int i;
    unsigned __int64 reis = 1; // 1. Reiskorn
    //cout << sizeof(__int64) << endl;

    //In der folgenden for-Schleife werden die Reiskörner pro Feld berechnet
    for (i = 2; i <= 64; i++)
    {
        reis = reis * 2; // alle weiteren Reiskörner
        cout << i << "tes Feld " << reis << endl;
    }
    // Beim letzten Wert kippt das Ergebnis bereits in den negativen Bereich,
    // wenn man unsigned weglässt. Die Zahl jedoch stimmt.
}

void transistorwiderstand(){
    double rc = 156, rl = 96.7;
    double h = rc + rl;
    double z = rc * rl / h;

    cout.precision(8); // 2 Vorkomma und 2 Nachkommastellen
    cout << z << endl;
}

void quadratischeGleichung(){
    double a, b, c, x1, x2;
    a = 5;
    b = 19;
    c = 14;

    double d = b * b - 4 * a * c;
```

```
        cout << "Diskriminante " << d << endl;
        if (d == 0)
        {
            x1 = (-1) * b / 2 / a;
            cout << "Eine Loesung: " << x1 << endl;
        }
        else if (d > 0)
        {
            double w = sqrt(d);
            x1 = ((-1) * b + w) / 2 / a;
            x2 = ((-1) * b - w) / 2 / a;
            cout << "1. Loesung: " << x1 << endl;
            cout << "2. Loesung: " << x2 << endl;
        }
        else
        {
            cout << "Keine Lösung" << endl;
        }
    }
}

void parallelschwingkreis(){
    double f, l, c;
    l = 0.4/1000;    // 0.4 Milli Henry
    c = 0.24/1000000000; // 240 Pico Farad

    const double PI = 3.14159265358979323846;
    double h = 2 * PI * sqrt(l * c);
    // Hilfsvariable h

    f = 1.0 / h;
    cout << "Frequenz f=" << f/1000 << " Kilo Hertz";
}

int _tmain(int argc, _TCHAR* argv[])
{
    //ascii();
    //kopfstand();
    //asciifolge();
    //reis();
    //transistorwiderstand();
    //quadratischeGleichung();
    parallelschwingkreis();
    getchar();
    return 0;
}
```