

```

// W11.cpp : Defines the entry point for the console application.
//          Formatowanie wejscia/wyjscia (funkcje i flagi klasy ios)

#include "stdafx.h"
#include <iostream>
#include <cmath>
using namespace std;

template <typename stream_T, typename stream_T1>
void showflags(stream_T &stream, stream_T1 &ostream);

int _tmain(int argc, _TCHAR* argv[])
{
    double a = 3.141592;
    int i = -24;
    char str[256] = " abcdef ";

    showflags(cout, cout);

    //dostan domyslne format flag settings flagdefaults
    ios::fmtflags flagdefault = cout.flags();

    cout << "default\n";
    cout << "|" << a << "|" << i << "|" << str << "|\n";

    cout << "width = 16" << endl;
    cout << "|"; cout.width(16); cout << a; cout << "|";
    cout.width(16); cout << i << "|";
    cout.width(16); cout << str << "|\n";

    cout << "width = 16 left\n";
    cout.setf(ios::left);
    cout << "|"; cout.width(16); cout << a; cout << "|";
    cout.width(16); cout << i << "|";
    cout.width(16); cout << str << "|\n";

    cout << "showpos width = 16\n";
    cout.setf(ios::showpos);
    cout << "|"; cout.width(16); cout << a; cout << "|";
    cout.width(16); cout << i << "|";
    cout.width(16); cout << str << "|\n";
}

```

```

cout << "showpos internal width = 16\n";
cout.unsetf(ios::left);
cout.setf(ios::showpos | ios::internal);
cout << "|"; cout.width(16); cout << a; cout << " | ";
cout.width(16); cout << i << " | ";
cout.width(16); cout << str << " |\n";

cout << "showpos right width = 16\n";
cout.unsetf(ios::internal);
cout.setf(ios::showpos | ios::right);
cout << "|"; cout.width(16); cout << a; cout << " | ";
cout.width(16); cout << i << " | ";
cout.width(16); cout << str << " |\n";

//ustaw format scientific
cout.setf(ios::scientific | ios::uppercase);
cout << "scientific | uppercase\n";
cout << "| " << a << " | " << i << " | " << str << " |\n";

//rezegnuj z flagi scientific
cout.unsetf(ios::scientific);
cout << "!scientific\n";
cout << "| " << a << " | " << i << " | " << str << " |\n";

//ustaw domyslne format flag settings flagdefaults
cout << "ustaw flagi domyslne\n";
cout.flags(flagdefault);
cout << "| " << a << " | " << i << " | " << str << " |\n";

streamsize default_prec = cout.precision();
cout.precision(3);
cout.setf(ios::left | ios::fixed);
cout << "TABELA LOGARYFMOW OD 2 DO 100\n\n";
cout << "x          ln(x)      log10(x)\n";

for(i=2; i<=100; i++)
{
    if(i<=20)
    {
        cout.fill('_');
    }
}

```

```

        else if(i > 20)
        {
            cout.precision(default_prec);
            cout.fill(' ');
        }
        cout.width(10); //po kazdym wykonaniu cout szerokosc pola automatycznie
                        //pozostaje przestawiona do wartosci domyslnej

        cout << i;
        cout.width(10);
        cout << log((double)i);
        cout.width(10);
        cout << log10((double)i);
        cout << "\n";
    }

    //ustaw domyslne format flag settings flagdefaults
    cout.flags(flagdefault);

    system("pause");
    return 0;
}

template <typename stream_T, typename stream_T1>
void showflags(stream_T &stream, stream_T1 &outstream)
{
    ios::fmtflags f;
    f = stream.flags();

    if(f & ios::skipws)
        outstream << "skipws"      1\n";
    else
        outstream << "skipws"      0\n";

    if(f & ios::left)
        outstream << "left"        1\n";
    else
        outstream << "left"        0\n";

    if(f & ios::right)
        outstream << "right"       1\n";
    else
        outstream << "right"       0\n";
}

```

```
if(f & ios::internal)
    ostream << "internal"    1\n";
else
    ostream << "internal"    0\n";

if(f & ios::dec)
    ostream << "dec"        1\n";
else
    ostream << "dec"        0\n";

if(f & ios::oct)
    ostream << "oct"        1\n";
else
    ostream << "oct"        0\n";

if(f & ios::oct)
    ostream << "oct"        1\n";
else
    ostream << "oct"        0\n";

if(f & ios::hex)
    ostream << "hex"        1\n";
else
    ostream << "hex"        0\n";

if(f & ios::showbase)
    ostream << "showbase"   1\n";
else
    ostream << "showbase"   0\n";

if(f & ios::showpoint)
    ostream << "showpoint"  1\n";
else
    ostream << "showpoint"  0\n";

if(f & ios::showpos)
    ostream << "showpos"    1\n";
else
    ostream << "showpos"    0\n";

if(f & ios::uppercase)
```

```
        ostream << "uppercase      1\n";
else
    ostream << "uppercase      0\n";

if(f & ios::scientific)
    ostream << "scientific    1\n";
else
    ostream << "scientific    0\n";

if(f & ios::fixed)
    ostream << "fixed          1\n";
else
    ostream << "fixed          0\n";

if(f & ios::unitbuf)
    ostream << "unitbuf       1\n";
else
    ostream << "unitbuf       0\n";

if(f & ios::boolalpha)
    ostream << "boolalpha     1\n";
else
    ostream << "boolalpha     0\n";

ostream << "\n";
}
```