

```

/* The following code example is taken from the book
 * "The C++ Standard Library – A Tutorial and Reference, 2nd Edition"
 * by Nicolai M. Josuttis, Addison-Wesley, 2012
 *
 * (C) Copyright Nicolai M. Josuttis 2012.
 * Permission to copy, use, modify, sell and distribute this software
 * is granted provided this copyright notice appears in all copies.
 * This software is provided "as is" without express or implied
 * warranty, and with no claim as to its suitability for any purpose.
 */
#include <locale>
#include <iostream>
#include <iterator>
#include <exception>
#include <cstdlib>

int main()
{
    // create copy of current global locale
    std::locale loc;

    // iterator to read from standard input
    typedef std::istreambuf_iterator<char> InIt;
    InIt beg = InIt(std::cin);
    InIt end = InIt();

    // stream which defines input format
    std::ios_base& fmt = std::cin;

    // declare output arguments
    std::ios_base::iostate err;
    float value;

    // get numeric input facet of the locale loc
    std::num_get<char, InIt> const& ng
        = std::use_facet<std::num_get<char, InIt> >(loc);

    // read value with numeric input facet
    ng.get(beg, end, fmt, err, value);

    // print value or error message
    if (err == std::ios_base::goodbit) {
        std::cout << "value: " << value << '\n';
    }
    else if (err == std::ios_base::eofbit) {
        std::cout << "value: " << value << " (EOF encountered)\n";
    }
    else if (err & std::ios_base::badbit) {
        std::cerr << "fatal error while reading numeric value\n";
        return EXIT_FAILURE;
    }
    else if (err & std::ios_base::failbit) {
        std::cerr << "format error while reading numeric value\n";
        return EXIT_FAILURE;
    }
}

```