

```

/* 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 <istream>
#include <limits>

class ignoreLine
{
private:
    int num;
public:
    explicit ignoreLine (int n=1) : num(n) {

        template <typename charT, typename traits>
        friend std::basic_istream<charT, traits>&
        operator>> (std::basic_istream<charT, traits>& strm,
                    const ignoreLine& ign)
        {
            // skip until end-of-line num times
            for (int i=0; i<ign.num; ++i) {
                strm.ignore(std::numeric_limits<std::streamsize>::max(),
                           strm.widen('\n'));
            }

            // return stream for concatenation
            return strm;
        }
    };
};

```