

Search:

Go

Not logged in

Reference

<istream>

istream

seekg

register

log in

C++

Information
Tutorials
Reference
Articles
Forum

Reference

Library:
Containers:
Input/Output:
 <fstream>
 <iomanip>
 <ios>
 <iosfwd>
 <iostream>
 <istream>
 <ostream>
 <sstream>
 <streambuf>
Multi-threading:
Other:

<istream>

class templates:
 basic_istream
 basic_iostream
classes:
 istream
 istream
 wistream
 wostream
manipulators:
 ws

istream

istream::istream
istream::~istream
member classes:
 istream::sentry
member functions:
 istream::gcount
 istream::get
 istream::getline
 istream::ignore
 istream::operator>>
 istream::peek
 istream::putback
 istream::read
 istream::readsome
 istream::seekg
 istream::sync
 istream::tellg
 istream::unget
non-member overloads:
 operator>> (istream)
protected members:
 istream::operator=
 istream::swap

public member function

std::istream::seekg

<istream> <iostream>

```
(1) istream& seekg (streampos pos);
(2) istream& seekg (streamoff off, ios_base::seekdir way);
```

Set position in input sequence

Sets the position of the next character to be extracted from the input stream.

Internally, the function accesses the input sequence by first constructing a [sentry](#) object (with *noskipws* set to true). Then (if good), it calls either [pubseekpos](#) (1) or [pubseekoff](#) (2) on its associated *stream buffer* object (if any). Finally, it destroys the *sentry* object before returning.

C++98

C++11



If the *eofbit* flag is set before the call, the function fails (sets *failbit* and returns).

Calling this function does not alter the value returned by [gcount](#).

Parameters

pos

New absolute position within the stream (relative to the beginning).
streampos is an *fpos* type (it can be converted to/from integral types).

off

Offset value, relative to the way parameter.
streamoff is an offset type (generally, a signed integral type).

way

Object of type *ios_base::seekdir*. It may take any of the following constant values:

value	offset is relative to...
<i>ios_base::beg</i>	beginning of the stream
<i>ios_base::cur</i>	current position in the stream
<i>ios_base::end</i>	end of the stream

Return Value

The *istream* object (*this).

Errors are signaled by modifying the *internal state flags*:

flag	error
<i>eofbit</i>	-
<i>failbit</i>	Either the construction of <i>sentry</i> failed, or the internal call to pubseekpos (1) or pubseekoff (2) failed (i.e., either function returned -1).
<i>badbit</i>	Another error occurred on the stream (such as when the function catches an exception thrown by an internal operation). When set, the integrity of the stream may have been affected.

Multiple flags may be set by a single operation.

If the operation sets an *internal state flag* that was registered with member *exceptions*, the function throws an exception of member type *failure*.

Example

```
1 // read a file into memory
2 #include <iostream>    // std::cout
3 #include <fstream>    // std::ifstream
4
5 int main () {
6     std::ifstream is ("test.txt", std::ifstream::binary);
7     if (is) {
8         // get length of file:
9         is.seekg (0, is.end);
10        int length = is.tellg();
11        is.seekg (0, is.beg);
12
13        // allocate memory:
14        char * buffer = new char [length];
15
16        // read data as a block:
17        is.read (buffer,length);
```

```
18
19     is.close();
20
21     // print content:
22     std::cout.write (buffer,length);
23
24     delete[] buffer;
25 }
26
27 return 0;
28 }
```

In this example, `seekg` is used to move the position to the end of the file, and then back to the beginning.

● Data races

Modifies the stream object.

Concurrent access to the same stream object may cause data races.

● Exception safety

Basic guarantee: if an exception is thrown, the object is in a valid state.

It throws an exception of member type `failure` if the resulting *error state flag* is not `goodbit` and member `exceptions` was set to throw for that state.

Any exception thrown by an internal operation is caught and handled by the function, setting `badbit`. If `badbit` was set on the last call to `exceptions`, the function rethrows the caught exception.

🔗 See also

<code>istream::tellg</code>	Get position in input sequence (public member function)
<code>ostream::seekp</code>	Set position in output sequence (public member function)