C++
Information
Tutorials
Reference
Articles
Forum

Reference
C library:
<cassert> (assert.h)
<cctype> (ctype.h)
<cerrno> (errno.h)
<cfenv> (fenv.h)
<cfloat> (float.h)
<cinttypes> (inttypes.h)
<ciso646> (iso646.h)
<climits> (limits.h)
<clocale> (locale.h)
<cmath> (math.h)
<csetjmp> (setjmp.h)
<csignal> (signal.h)
<cstdarg> (stdarg.h)
<cstdbool> (stdbool.h)
<cstddef> (stddef.h)
<cstdint> (stdint.h)
<cstdio> (stdio.h)
<cstdlib> (stdlib.h)
<cstring> (string.h)
<ctgmath> (tgmath.h)
<ctime> (time.h)
<cuchar> (uchar.h)
<cwchar> (wchar.h)
<cwctype> (wctype.h)
Containers:
Input/Output:
Multi-threading:
Other:

<cstdlib> (stdlib.h)
functions:
abort
abs
atexit
atof
atoi
atol
atoll
at_quick_exit
bsearch
calloc
div
exit
free
getenv
labs
ldiv
llabs
lldiv
malloc
mblen
mbstowcs
mbtowc
qsort
quick_exit
rand
realloc
srand
strtod
strtof
strtol
strtold
strtoll
strtoul
strtoull
system
wcstombs
wctomb
_Exit
functions (non-standard):
itoa
types:
div_t
ldiv_t
lldiv_t
size_t

function

atoi

<cstdlib>

```
int atoi (const char * str);
```

Convert string to integer

Parses the C-string *str* interpreting its content as an integral number, which is returned as a value of type `int`.

The function first discards as many whitespace characters (as in `isspace`) as necessary until the first non-whitespace character is found. Then, starting from this character, takes an optional initial *plus* or *minus* sign followed by as many base-10 digits as possible, and interprets them as a numerical value.

The string can contain additional characters after those that form the integral number, which are ignored and have no effect on the behavior of this function.

If the first sequence of non-whitespace characters in *str* is not a valid integral number, or if no such sequence exists because either *str* is empty or it contains only whitespace characters, no conversion is performed and zero is returned.

Parameters

`str`
C-string beginning with the representation of an integral number.

Return Value

On success, the function returns the converted integral number as an `int` value.
If the converted value would be out of the range of representable values by an `int`, it causes *undefined behavior*. See `strtol` for a more robust cross-platform alternative when this is a possibility.

Example

```
/* atoi example */
#include <stdio.h>      /* printf, fgets */
#include <stdlib.h>     /* atoi */

int main ()
{
  int i;
  char buffer[256];
  printf ("Enter a number: ");
  fgets (buffer, 256, stdin);
  i = atoi (buffer);
  printf ("The value entered is %d. Its double is %d.\n",i,i*2);
  return 0;
}
```

Output:

```
Enter a number: 73
The value entered is 73. Its double is 146.
```

Data races

The array pointed by *str* is accessed.

Exceptions (C++)

No-throw guarantee: this function never throws exceptions.

If *str* does not point to a valid C-string, or if the converted value would be out of the range of values representable by an `int`, it causes *undefined behavior*.

See also

atol	Convert string to long integer (function)
atof	Convert string to double (function)
strtol	Convert string to long integer (function)

macro constants:
EXIT_FAILURE
EXIT_SUCCESS
MB_CUR_MAX
NULL
RAND_MAX