

3.6 Random Sources (`random_source`)

1. Definition

An instance of type `random_source` is a random source. It allows to generate uniformly distributed random bits, characters, integers, and doubles. It can be in either of two modes: In bit mode it generates a random bit string of some given length p ($1 \leq p \leq 31$) and in integer mode it generates a random integer in some given range $[low..high]$ ($low \leq high < low + 2^{31}$). The mode can be changed any time, either globally or for a single operation. The output of the random source can be converted to a number of formats (using standard conversions).

```
#include < LEDA/random_source.h >
```

2. Creation

`random_source S;` creates an instance S of type `random_source`, puts it into bit mode, and sets the precision to 31.

`random_source S(int p);`
creates an instance S of type `random_source`, puts it into bit mode, and sets the precision to p ($1 \leq p \leq 31$).

`random_source S(int low, int high);`
creates an instance S of type `random_source`, puts it into integer mode, and sets the range to $[low..high]$.

3. Operations

`unsigned long S.get()` returns a random unsigned integer of maximal precision (32 bits on 32-bit systems and 64 bits on 64-bit systems).

`void S.set_seed(int s)` resets the seed of the random number generator to s .

`void S.set_range(int low, int high)`
sets the mode to integer mode and changes the range to $[low..high]$.

`int S.set_precision(int p)` sets the mode to bit mode, changes the precision to p bits and returns previous precision.

`int S.get_precision()` returns current precision of S .

<i>random_source</i> & <i>S</i> >> <i>char</i> & <i>x</i>	extracts a character <i>x</i> of default precision or range and returns <i>S</i> , i.e., it first generates an unsigned integer of the desired precision or in the desired range and then converts it to a character (by standard conversion).
<i>random_source</i> & <i>S</i> >> <i>unsigned char</i> & <i>x</i>	extracts an unsigned character <i>x</i> of default precision or range and returns <i>S</i> .
<i>random_source</i> & <i>S</i> >> <i>int</i> & <i>x</i>	extracts an integer <i>x</i> of default precision or range and returns <i>S</i> .
<i>random_source</i> & <i>S</i> >> <i>long</i> & <i>x</i>	extracts a long integer <i>x</i> of default precision or range and returns <i>S</i> .
<i>random_source</i> & <i>S</i> >> <i>unsigned int</i> & <i>x</i>	extracts an unsigned integer <i>x</i> of default precision or range and returns <i>S</i> .
<i>random_source</i> & <i>S</i> >> <i>unsigned long</i> & <i>x</i>	extracts a long unsigned integer <i>x</i> of default precision or range and returns <i>S</i> .
<i>random_source</i> & <i>S</i> >> <i>double</i> & <i>x</i>	extracts a double precision floating point number <i>x</i> in $[0, 1]$, i.e, $u/(2^{31} - 1)$ where <i>u</i> is a random integer in $[0..2^{31} - 1]$, and returns <i>S</i> .
<i>random_source</i> & <i>S</i> >> <i>float</i> & <i>x</i>	extracts a single precision floating point number <i>x</i> in $[0, 1]$, i.e, $u/(2^{31} - 1)$ where <i>u</i> is a random integer in $[0..2^{31} - 1]$, and returns <i>S</i> .
<i>random_source</i> & <i>S</i> >> <i>bool</i> & <i>b</i>	extracts a random boolean value (true or false).
<i>int</i> <i>S</i> ()	returns an integer of default precision or range.
<i>int</i> <i>S</i> (<i>int prec</i>)	returns an integer of supplied precision <i>prec</i> .
<i>int</i> <i>S</i> (<i>int low</i> , <i>int high</i>)	returns an integer from the supplied range [<i>low</i> .. <i>high</i>].