characters (such as carriage returns and linefeeds) occurring therein; in the display such characters are replaced by spaces. For example, try `< 8 32 $ a.`

Floor `<.` 0 0 0 Lesser Of (Min)

`<.y` gives the floor of `y`, that is, the largest integer less than or equal to `y`. Thus:
`   <. 4.6 4 _4 _4.6`
`4 4 _4 _5`
 	
The implied comparison with integers is tolerant, as discussed under Equal (`=`), and is controlled by `<.!.t` . See below for complex arguments.
	`x<.y` is the lesser of `x` and `y`. For example:
	`   3 <. 4 _4`
	`3 _4`
	`   <./7 8 5 9 2`
	`2`
	`   <./\7 8 5 9 2`
	`7 7 5 5 2`

For a complex argument, the definition of `<.` is modelled by:

```
   floor=: j./@(ip+(c2>c1),c1+:c2)
   '`c1 c2 fp ip'=:(1:>+/@fp)`(>:/@fp)`(+.-ip)`(<.@+.)
```

As developed by McDonnell [10], this function has the following properties:

Convexity: If `(<.z1)=(<.z2)` and `z3` lies on the line between `z1` to `z2`, then `(<z3)=(<z1)`.
Translatability: If `z4` is a Gaussian integer, then `(z4+<.z5)=(<.z4+z5)`.
Compatibility: `(<.x j.0)=((<.x)j.0)` and `(<.0 j.x)=(0 j.(<.x))`

The function `<.` can be viewed as a tiling by rectangles of unit area, all arguments within a rectangle sharing the same floor. One rectangle has vertices at `1j0` and `0j1`, with the other side passing through the origin. Rectangles along successive diagonals are displaced by one-half the length.

The phrase `j./@ip` “floors” the individual parts of a complex argument. Moreover, the floor `<.y` is equivalent to `->.-y` . In other words, it is the dual of ceiling with respect to (that is, under) arithmetic negation: `<.` ↔ `>.&.-` and `>.` ↔ `<.&.-` . Thus:

```
   (>.&.- ; <.) 4.6 4 _4 _4.6
+---------+---------+
|4 4 _4 _5|4 4 _4 _5|
+---------+---------+
```

Continued

Floor <. 0 0 0 Lesser Of (Min)

Continued

The expression `<.x+0.5` gives the integer nearest to the real argument `x`, and `<.z+0.5j0.5` gives the Gaussian integer nearest to `z`. The number of digits needed to represent an integer is given by one plus the floor of its base ten logarithm:

```
   a ,. (,. 1:+<.) 10^. a=: 9 10 11 99 100 101
  9 0.954243 1
 10        1 2
 11  1.04139 2
 99  1.99564 2
100        2 3
101  2.00432 3
```