Treat complex<T> as Unit in Stream Output

In section 26.2.5, **`complex`** non-member operations, change

```
template<class T>
     ostream& operator<<(ostream& os, complex x);
```

Returns: `os << ‘(’ << x.real() << ‘,’ << x.imag() << ‘)’ .`

to

A string is constructed in the pattern (r,i) where r is the real part and i the imaginary part of the complex object x in accordance with the relevant format flags in the supplied ostream. There are no intervening spaces. The string is then passed to the supplied ostream where flags involving width and justification are applied. Non-optimized implementation:

```
{
     ostringstream ost;
     ost.setf(os.flags());
     ost << ‘(‘ << x.real() << ‘,’ << x.imag() << ‘)’ ;
     return os << ost.str() ;
}
```

Reason:

Under the current rules, the user’s “width” setting will apply to the opening paren rather than the expression as a whole.