

```
1 // Sobrecarga del operador <<
2
3 ostream & operator <<(ostream &out, const struct St &st){
4     out.precision(2);
5     out<<fixed;
6     out<<setw(10)<<st.campo<<...;
7     return out;
8 }
9
10 tipol operator <<(tipo2 param1, tipo3 param){
11     // tipol, tipo2, tipo3 pueden ser cualquier tipo de dato,
12     // por ejemplo ostream, ostream &, ofstream, o cualquiera
13     // otro tipo de dato, siempre y cuando el juego de parámetros
14     // no corresponda a algún juego ya establecido que produzca
15     // un error de ambigüedad.
16     // La orden return es opcional y depende del tipo de dato
17 }
18
19 // Sobrecarga del operador >>
20
21 istream & operator <<(istream &in, const struct St &st){
22     in>>st.campo>>...;
23
24     return out;
25 }
26
27 tipol operator <<(tipo2 param1, tipo3 param){
28     // tipol, tipo2, tipo3 pueden ser cualquier tipo de dato,
29     // por ejemplo istream, istream &, ifstream, o cualquiera
30     // otro tipo de dato, siempre y cuando el juego de parámetros
31     // no corresponda a algún juego ya establecido que produzca
32     // un error de ambigüedad.
33     // La orden return es opcional y depende del tipo de dato
34 }
35
36 // Sobrecarga del operador ++
37 tipol operator ++(tipo2 param){
38     // Tiene un solo parametro, se usa de esta forma: ++dato;
39     // NO SE PUEDE USAR dato++ con este modelo
40     // tipol, tipo2 pueden ser cualquier tipo de dato, siempre
41     // y cuando el parámetro no corresponda a algún tipo de dato
42     // ya establecido que produzca un error de ambigüedad.
43     // La orden return es opcional y depende del tipo de dato
44 }
```