Multi Sorted Species

$F : \mathbb{B} \longrightarrow \mathbb{B}$

List(L:LT):CS(L)

CS(L:LT):Cat = LT+..

$M : \mathbb{B}^k \longrightarrow \mathbb{B}$

$(F \circ G)(L)$

$G_1 .. G_k : \mathbb{B}^m \to \mathbb{B}$

$M(G_1(L_1 \ldots L_m), \ldots G_k(L_1 \ldots L_m))$

$M : \mathbb{B}^k \to \mathbb{B}$

$M(T : \mathrm{Tuple}(MCS(?))^{PT}) : MCS(\ldots)^{PT} = =$

$prj : I \to T_i$

$prj : (\%_0, i : I) \to T\ i$

$[\quad]$

$M(Int, Str)$

$(T, List) \mapsto (List\ T_1, \ldots List\ T_k)$

$(Int, Str) \longmapsto (List\ Int, List\ Str)$

```
T : Tuple(Type) == (I, S, F)

F(T : Tuple(Type), U : M(T)) : ..... == {
  for i in 1..length(T) repeat {
    tj : TI
    where { j := prt(F, j)
            j == i;
          }
    TJ == T.j }
}

Tth(S : Type) : with {
  typ : (Int, Str) -> %
}

structures([1,2,3], ["a", "b"])(1, "a")
```

T : Tuple(Ring)

C == Cross(T)

C : Tuple Cross(Ring, Monoid) = (Int, Float)

Domain — element

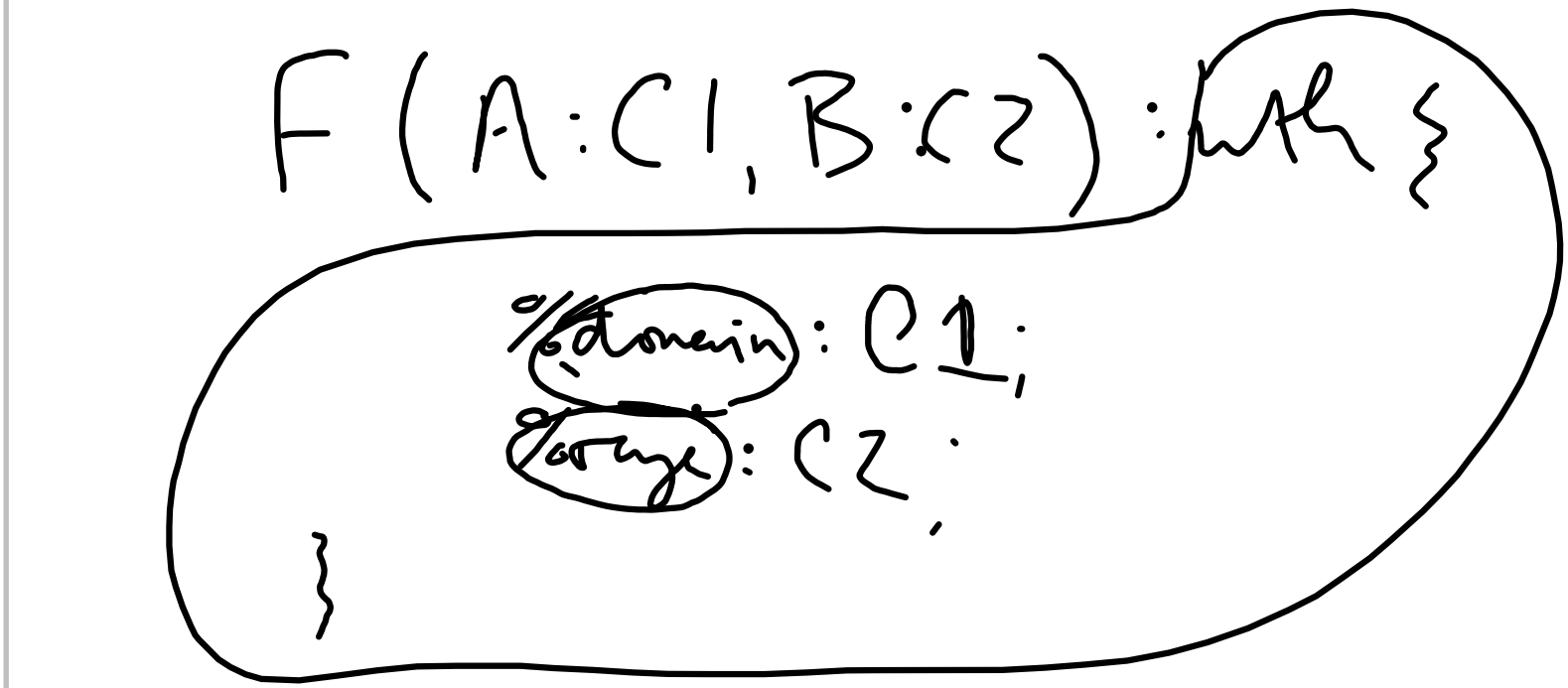
t : Type == Cross(C)

f(T1 : Ring, T2 : Ring) : ... == .....

f(C)

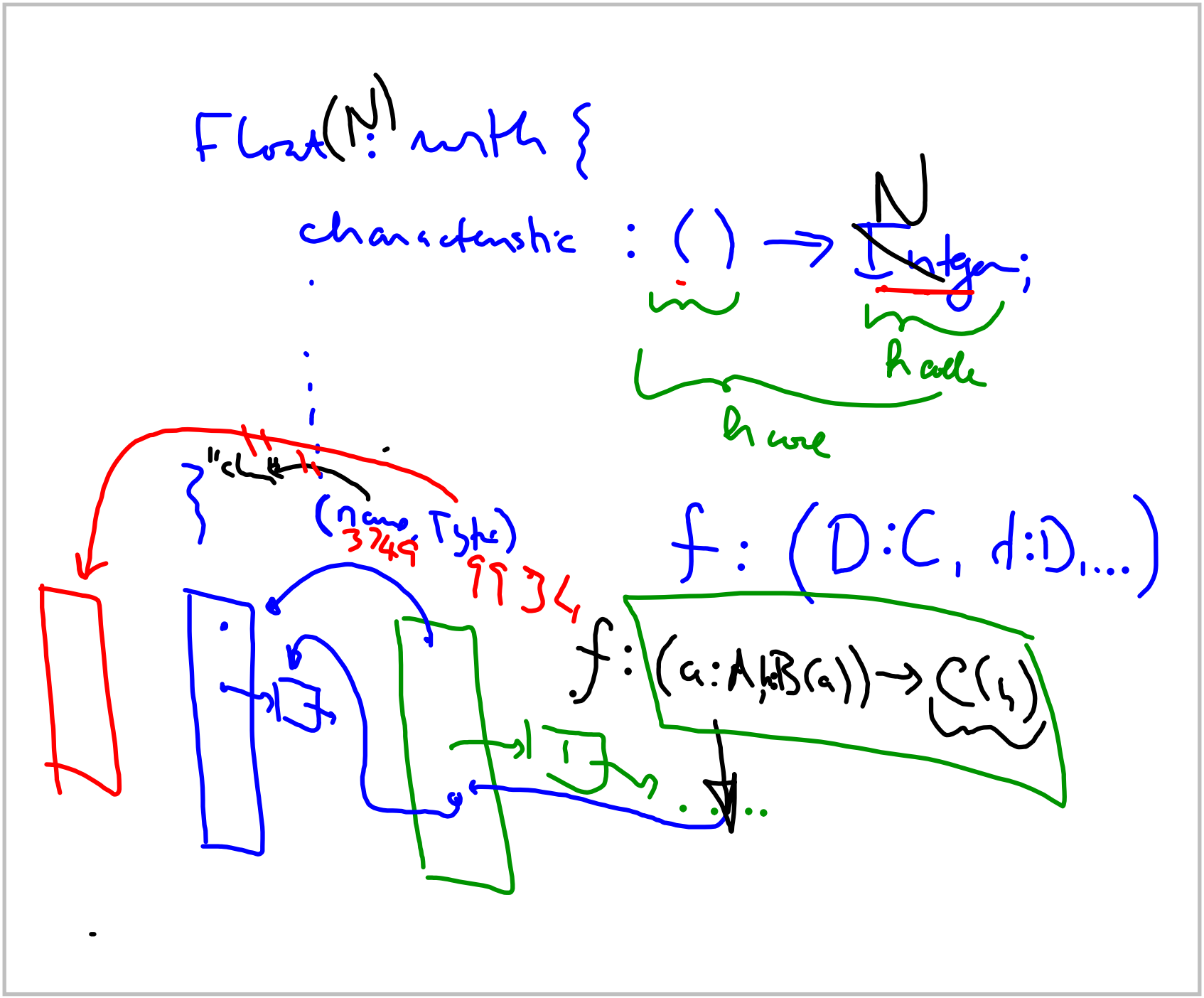
```
template <typename C>
class Algebra
{
public:
        typedef   list<C> Carrier_t;
          :
          ;


};
```



==TensorMapping Cat.

R : with {
Type
f : % → %


Float(N) with {
characteristic : () → N
Integer;
Read
Write
"c
(name 3749 Type 9934)
f : (D:C, d:D,...)
f : (a:A, b:B(a)) → C(a,b)

$$\frac{\Gamma \vdash a : C(A, B)}{\mathrm{fst}\ a : A \quad \wedge \quad \mathrm{snd}\ a : B}$$

$$\frac{a : D(x : A, B(x)) \qquad C(a : A(x), b : B(x))}{\mathrm{fst}\ a : \ast \quad \text{in} \quad \mathrm{let}\ (u, v) = a \ \text{in}}$$

$u : A, v : B(x)$