5. SIGNAL FLOW GRAPHS

⇒ Prop :

Definition 5.2. A *prop* is a symmetric strict monoidal category $(\mathcal{C}, 0, +)$ for which $\mathrm{Ob}(\mathcal{C}) = \mathbb{N}$, the monoidal unit is $0 \in \mathbb{N}$, and the monoidal product on objects is given by addition.

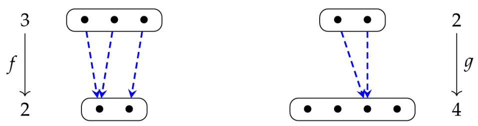
Note that each object n is the n-fold monoidal product of the object 1; we call 1 the *generating object*. Since the objects of a prop are always the natural numbers, to specify a prop P it is enough to specify five things:

(i) a set $\mathcal{C}(m, n)$ of morphisms $m \to n$, for $m, n \in \mathbb{N}$.
(ii) for all $n \in \mathbb{N}$, an identity map $\mathrm{id}_n : n \to n$.
(iii) for all $m, n \in \mathbb{N}$, a symmetry map $\sigma_{m,n} : m + n \to n + m$.
(iv) a composition rule: given $f : m \to n$ and $g : n \to p$, a map $(f \mathbin{;} g) : m \to p$.
(v) a monoidal product on morphisms: given $f : m \to m'$ and $g : n \to n'$, a map $(f + g) : m + n \to m' + n'$.

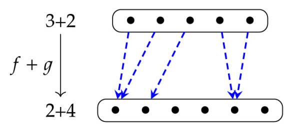
Eg: FINSET:
morphism $f : m \to n$ are function from $\underline{m} = \{1, 2, \ldots, m\}$ to $\underline{n} = \{1, 2, \ldots, n\}$ ⇒ id, sym & comp. are same as for functions. monoidal product is "disjoint union of functions" (+) $f : m \to m'$ & $n \to n'$; $f + g : m + n \to m' + n'$

Eg for this is :

1. Below we draw a morphism $f : 3 \to 2$ and a morphism $g : 2 \to 4$ in **FinSet**:



2. Here is a picture of $f + g$



* There is a subcategory of FINSET : BIJ, as morph. exist in BIJ only if $m = n$.

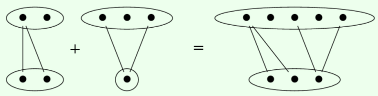
Other examples :

Example 5.7. The compact closed category **Corel**, in which the morphisms $f\colon m \to n$ are partitions on $\underline{m} \sqcup \underline{n}$ (see Example 4.61), is a prop.

Example 5.8. There is a prop **Rel** for which morphisms $m \to n$ are relations, $R \subseteq \underline{m} \times \underline{n}$. The composition of R with $S \subseteq \underline{n} \times \underline{p}$ is

$$R \mathbin{;} S := \{(i,k) \in \underline{m} \times \underline{p} \mid \exists (j \in \underline{n}).\, (i,j) \in R \text{ and } (j,k) \in S\}.$$

The monoidal product is relatively easy to formalize using universal properties,[3] but one might get better intuition from pictures:



⇒ Prop functor : $F : \mathcal{C} \to \mathcal{D}$

(a) $F(n) = n \quad \forall\, n \in ob(\mathcal{C}) = ob(\mathcal{D}) = \mathbb{N}$

(b) $\forall\, f : m_1 \to m_2$ & $g : n_1 \to n_2$ in $\mathcal{C}$,

$$F(f) + F(g) = F(f + g)$$

⇒ Port Graphs :

Definition 5.13. For $m, n \in \mathbb{N}$, an (m,n)-*port graph* (V, in, out, ι) is specified by

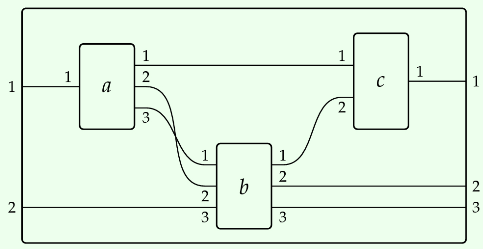
(i) a set V, elements of which are called *vertices*,

(ii) functions $in, out\colon V \to \mathbb{N}$, where $in(v)$ and $out(v)$ are called the *in degree* and *out degree* of each $v \in V$, and

(iii) a bijection $\iota\colon \underline{m} \sqcup O \xrightarrow{\cong} I \sqcup \underline{n}$, where $I = \{(v,i) \mid v \in V,\ 1 \le i \le in(v)\}$ is the set of *vertex inputs*, and $O = \{(v,i) \mid v \in V,\ 1 \le i \le out(v)\}$ is the set of *vertex outputs*.

This data must obey the following acyclicity condition. First, use the bijection ι to construct the graph with vertices V and with an arrow $e^{u,i}_{v,j}\colon u \to v$ for every $i, j \in \mathbb{N}$ such that $\iota(u,i) = (v,j)$; call it the *internal flow graph*. If the internal flow graph is acyclic—that is, if the only path from any vertex v to itself is the trivial path—then we say that (V, in, out, ι) is a port graph.

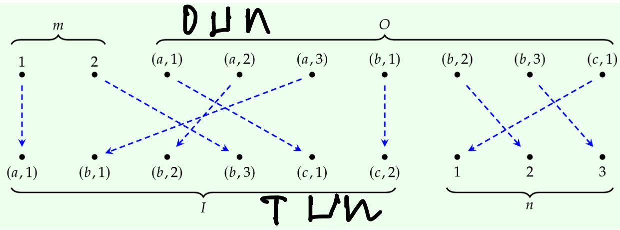
Example 5.14. Here is an example of a $(2,3)$-port graph, i.e. with $m = 2$ and $n = 3$:



(5.15)

The Bijection Map :

$O \sqcup n$



$I \sqcup n$

for in() & out() func, eg: in(a) = 1 & out(a) = 3

* Category PG whose morphisms are port graphs

for two pgs (morphisms)

(m,n) graph $= (V, in, out, \iota)$

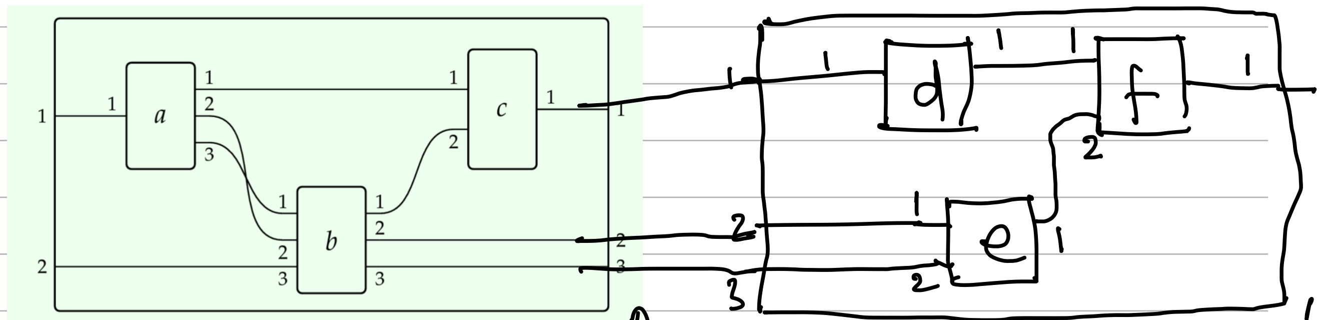
(n,p) graph $= (V', in', out', \iota')$

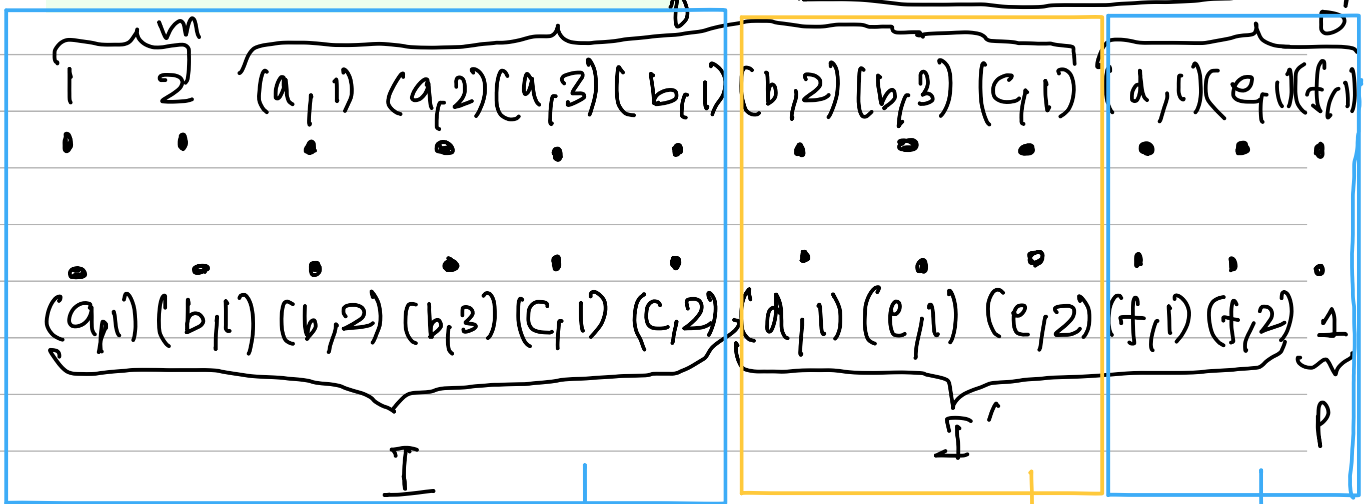
the composed graph (morph.) is :

(m,p) graph $= (V \sqcup V', in \sqcup in', out \sqcup out', \iota'')$

$\iota'' : \underline{m} \sqcup O \sqcup O' \to I \sqcup I' \sqcup \underline{p}$

$$\iota''(x) = \begin{cases} \iota(x) & \text{if } \iota(x) \in I \\ \iota'(\iota(x)) & \text{if } \iota(x) \in \underline{n} \\ \iota'(x) & \text{if } x \in O.' \end{cases}$$





Mapping same as $\iota(x)$

$\iota'(\iota(x))$

$\iota(x)$

Monoidal product: $G + G' = ((V \sqcup V'), [in, in'], [out, out'], (\iota \sqcup \iota'))$

⇒ Free Constructions :
↓
minimally - constrained

for a preorder : the minimal constraint will be the reflexive & transitive closure of a relation.
↳ because the other equalities doesn't matter in defining monotone maps

* So we call these properties as "universal properties" of a preorder $(P, \leq_P)$

⇒ Free category on graph :

the paths in G. (So **Free**(G) is a category that in a sense contains G and obeys no equations other than those that categories are forced to obey.)

Refer eg: 5.23

⇒ Free prop on a signature :

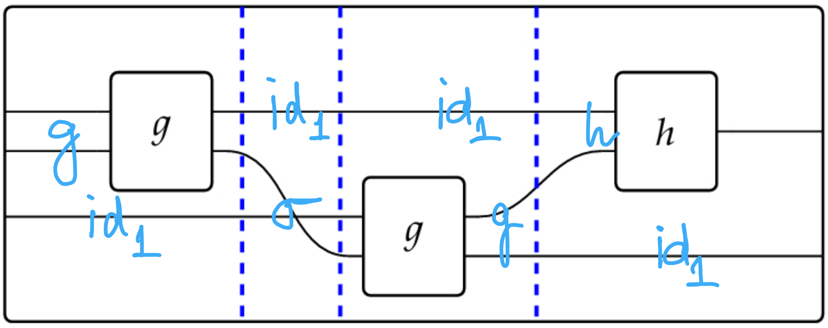
Definition 5.25. A *prop signature* is a tuple (G, s, t), where G is a set and $s, t \colon G \to \mathbb{N}$ are functions; each element $g \in G$ is called a *generator* and $s(g), t(g) \in \mathbb{N}$ are called its *in-arity and out-arity*. We often denote (G, s, t) simply by G, taking s, t to be implicit.

A *G-labeling* of a port graph $\Gamma = (V, in, out, \iota)$ is a function $\ell \colon V \to G$ such that the arities agree: $s(\ell(v)) = in(v)$ and $t(\ell(v)) = out(v)$ for each $v \in V$.

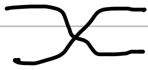
Define the *free prop on* G, denoted **Free**(G), to have as morphisms $m \to n$ all G-labeled (m, n)-port graphs. The composition and monoidal structure are just those for port graphs **PG** (see Eq. (5.17)); the labelings (the ℓ's) are just carried along.

Proposition 5.29. The free prop $\mathbf{Free}(G)$ on a signature (G, s, t) has the property that, for any prop $\mathcal{C}$, the prop functors $\mathbf{Free}(G) \to \mathcal{C}$ are in one-to-one correspondence with functions $G \to \mathcal{C}$ that send each $g \in G$ to a morphism $s(g) \to t(g)$ in $\mathcal{C}$.

Example :



σ : Swap Map



exp : $(g + id_1) ; (id_1 + \sigma) ; (id_1 + g) ; (h + id_1)$

⇒ Presentation :

Rough Definition 5.33. A *presentation* (G, s, t, E) for a prop is a set G, functions $s, t\colon G \to \mathbb{N}$, and a set $E \subseteq \mathrm{Expr}(G) \times \mathrm{Expr}(G)$ of pairs of G-generated prop expressions, such that e_1 and e_2 have the same arity for each $(e_1, e_2) \in E$. We refer to G as the set of generators and to E as the set of *equations* in the presentation.[6]

The prop $\mathcal{G}$ *presented* by the presentation (G, s, t, E) is the prop whose morphisms are elements in $\mathrm{Expr}(G)$, quotiented by both the equations $e_1 = e_2$ where $(e_1, e_2) \in E$, and by the axioms of symmetric strict monoidal categories.

⇒ Simplified Signal Flow Graphs :

* Rigs :

Used to represent a signal which can be multiplied (amplified) & added and the distributive law holds

Definition 5.36. A *rig* is a tuple $(R, 0, +, 1, *)$, where R is a set, $0, 1 \in R$ are elements, and $+, *\colon R \times R \to R$ are functions, such that

(a) $(R, +, 0)$ is a commutative monoid,
(b) $(R, *, 1)$ is a monoid,[8] and
(c) $a * (b + c) = a * b + a * c$ and $(a + b) * c = a * c + b * c$ for all $a, b, c \in R$.
(d) $a * 0 = 0 = 0 * a$ for all $a \in R$.

Examples :

① $(\mathbb{N}, 0, +, 1, *)$ ② $(\mathbb{B}, \text{false}, \vee, \text{true}, \wedge)$

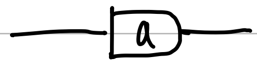
Generally, for any quantale $\mathcal{V} = (V, 0, \vee, I, \otimes)$

where 0 is empty join $0 = \vee \phi$

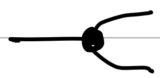
Example 5.40. If R is a rig and $n \in \mathbb{N}$ is any natural number, then the set $\mathrm{Mat}_n(R)$ of $(n \times n)$-matrices in R forms a rig. A matrix $M \in \mathrm{Mat}_n(R)$ is a function $M : \underline{n} \times \underline{n} \to R$. Addition $M + N$ of matrices is given by $(M + N)(i, j) := M(i, j) + N(i, j)$ and multiplication $M * N$ is given by $(M * N)(i, j) := \sum_{k \in \underline{n}} M(i, k) * N(k, j)$. The 0-matrix is $0(i, j) := 0$ for all $i, j \in \underline{n}$. Note that $\mathrm{Mat}_n(R)$ is generally not commutative.

Example 5.42. Any ring forms a rig. In particular, the real numbers $(\mathbb{R}, 0, +, 1, *)$ are a rig. The difference between a ring and rig is that a ring, in addition to all the properties of a rig, must also have additive inverses, or *negatives*. A common mnemonic is that a rig is a **ri**ng without **n**egatives.

Iconography :

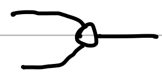


amplifies by a (scalar multi.)



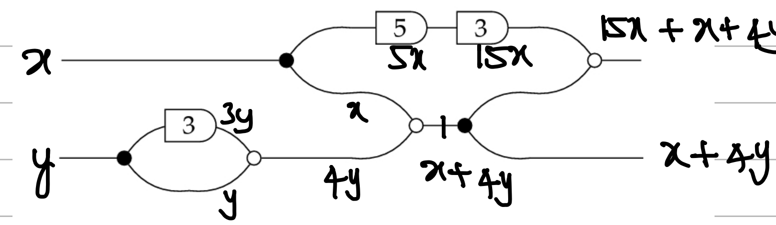
copy a signal

discard



add signals

Eg :



$15x + x + 4y = 16x + 4y$

Another way:

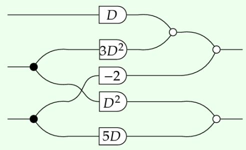
Example 5.44. This example is [illegible]me familiarity with differential equations. A linear system of differential equations provides a simple way to specify the movement of a particle. For example, consider a particle whose position (x, y, z) in 3-dimensional space is determined by the following equations:

$$\begin{aligned} \dot{x} + 3\ddot{y} - 2z &= 0 \\ \ddot{y} + 5\dot{z} &= 0 \end{aligned}$$

Using what is known as the Laplace transform, one can convert this into a linear system involving a formal variable D, which stands for "differentiate." Then the system becomes

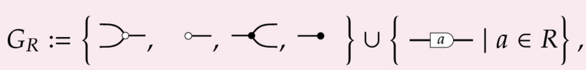
$$\begin{aligned} Dx + 3D^2y - 2z &= 0 \\ D^2y + 5Dz &= 0 \end{aligned}$$

which can be represented by the signal flow graph



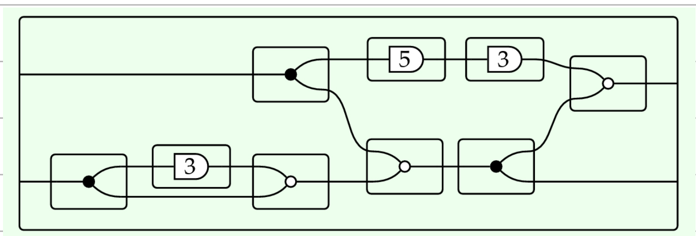
⇒

Definition 5.45. Let R be a rig (see Definition 5.36). Consider the set



and let $s, t\colon G_R \to \mathbb{N}$ be given by the number of dangling wires on the left and right of the generator icon respectively. A *simplified signal flow graph* is a morphism in the free prop $\mathbf{Free}(G_R)$ on this set G_R of generators. We define $\mathbf{SFG}_R := \mathbf{Free}(G_R)$.

To see similarities –



⇒ The prop of matrices over a rig :

Definition 5.50. Let R be a rig. We define the *prop of R-matrices*, denoted $\mathbf{Mat}(R)$, to be the prop whose morphisms $m \to n$ are the $(m \times n)$-matrices with values in R. Composition of morphisms is given by matrix multiplication as in Eq. (5.48). The monoidal product is given by the direct sum of matrices: given matrices $A\colon m \to n$ and $b\colon p \to q$, we define $A + B\colon m + p \to n + q$ to be the block matrix

$$\begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix}$$

where each 0 represents a matrix of zeros of the appropriate dimension ($m \times q$ and $n \times p$). We refer to any combination of multiplication and direct sum as a *interconnection* of matrices.

Eg: $A = \begin{pmatrix} 3 & 3 & 1 \\ 2 & 0 & 4 \end{pmatrix}$ $B = \begin{pmatrix} 2 & 5 & 6 & 1 \end{pmatrix}$

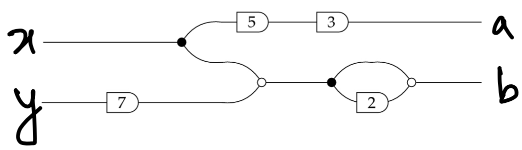
$$A + B = \begin{pmatrix} 3 & 3 & 1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 5 & 6 & 1 \end{pmatrix}$$

Signal Flow Graphs $\rightarrow$ Matrices

$SFG_R \longrightarrow Mat(R)$

$m \times n$ matrix

$m=2$



$n=2$

	a	b
x	15	3
y	0	21

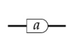
$$\begin{pmatrix} 15 & 3 \\ 0 & 21 \end{pmatrix}$$

$a = 15x + 0y$
$b = 3x + 21y$

We can say:

Proposition 5.54. Let g be a signal flow graph with m inputs and n outputs. The matrix $S(g)$ is the $(m \times n)$-matrix whose (i, j)-entry describes the amplification of the ith input that contributes to the jth output.

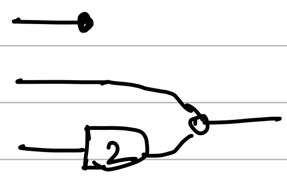
other eg.:

 $\mapsto \begin{pmatrix} 1 \end{pmatrix}$ and  $\mapsto \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ and 

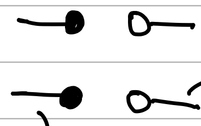
$\mapsto \begin{pmatrix} a \end{pmatrix}$

$$\begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} =$$

	a
x	0
y	1
z	2

$=$ 

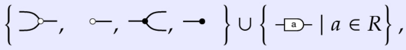
$$\begin{array}{c|cc} & a & b \\ \hline x & \multirow{2}{*}{$\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$} \\ y & \end{array} =$$



indicates $0 \rightarrow a$ or $0 \rightarrow b$ $(0 \rightarrow 1)$

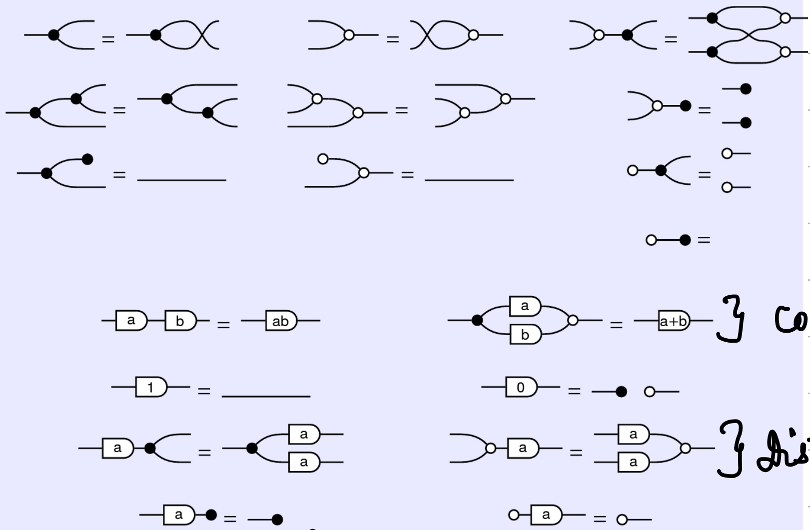
indicates $x \rightarrow 0$ or $y \rightarrow 0$ $(1 \rightarrow 0)$

Theorem 5.60. The prop $\mathbf{Mat}(R)$ is isomorphic to the prop with the following presentation. The set of generators is the set

$G_R :=$ 

the same as the set of generators for $\mathbf{SFG}_R$; see Definition 5.45.

We have the following equations for any $a, b \in R$:



} commutativity

} distributive laws of a rig

$a \times 0 = 0$ $\quad$ $0 \times a = 0$

⇒ Monoidal obj. in a monoidal category :

Definition 5.65. A *monoid object* (M, μ, η) in a symmetric monoidal category $(\mathcal{C}, I, \otimes)$ is an object M of $\mathcal{C}$ together with morphisms $\mu\colon M \otimes M \to M$ and $\eta\colon I \to M$ such that

(a) $(\mu \otimes \mathrm{id}) \mathbin{⨟} \mu = (\mathrm{id} \otimes \mu) \mathbin{⨟} \mu$ and

(b) $(\eta \otimes \mathrm{id}) \mathbin{⨟} \mu = \mathrm{id} = (\mathrm{id} \otimes \eta) \mathbin{⨟} \mu$.

A *commutative monoid object* is a monoid object that further obeys

(c) $\sigma_{M,M} \mathbin{⨟} \mu = \mu$.

where $\sigma_{M,M}$ is the swap map on M in $\mathcal{C}$. We often denote it simply by σ.

Eg: $(\mathbb{R}, +, 0)$ is a monoid obj. if $\mu \Rightarrow \mu(a,b) = a+b$ & $\eta = 0$

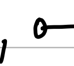
Ex 5.67

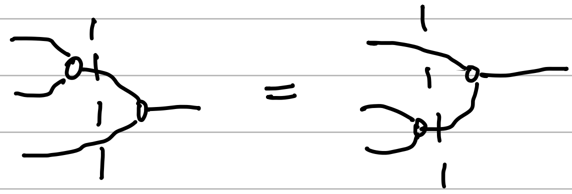
a)
$$\begin{tikzcd} (a,b,c) \arrow[r, mapsto, "(\mu\otimes\mathrm{id})"] \arrow[d, mapsto, "(\mathrm{id}\otimes\mu)"'] & (a+b,c) \arrow[d, mapsto, "\mu"] \\ (a,b+c) \arrow[r, mapsto, "\mu"'] & a+b+c \end{tikzcd}$$

b)
$$\begin{tikzcd} a \arrow[r, mapsto, "(\eta\otimes\mathrm{id})"] \arrow[d, mapsto, "(\mathrm{id}\otimes\eta)"'] \arrow[dr, mapsto, "\mathrm{id}_a"] & (0,a) \arrow[d, mapsto, "\mu"] \\ (a,0) \arrow[r, mapsto, "\mu"'] & a \end{tikzcd}$$

c)
$$\begin{tikzcd} (a,b) \arrow[r, mapsto, "\sigma"] \arrow[dr, mapsto, "\mu"'] & (b,a) \arrow[d, mapsto, "\mu"] \\ & a+b \end{tikzcd}$$

DIDN'T UNDERSTAND THIS...

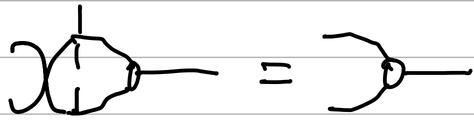
* (1,  , ) is a commutative monoid obj. in prop MAT(R)

because, ① 

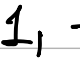
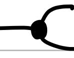
$\mu \otimes id \,;\, \mu \qquad = \qquad id \otimes \mu \,;\, \mu$

② 

$id \otimes \eta \,;\, \mu \qquad = \qquad id \qquad = \qquad \eta \otimes id \,;\, \mu$

③ 

$\sigma \,;\, \mu \qquad = \qquad \mu$

* (1,  , ) is a comm. monoidal obj. in $MAT(R)^{op}$

⇒ Behaviour of SFGs :

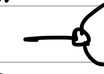
the SFG $g : m \to n$ has the behaviour

$$B(g) = \{ (x, S(g)(x)) \mid x \in R^m \} \subseteq R^m \times R^n$$

Mirror image of icon behaviour

$$B(g^{op}) = \{ (S(g)(x), x) \mid x \in R^m \} \subseteq R^n \times R^m$$

is called transposed Relation

B() $= \{ ((y, z), x) \mid x \in R, x = y + z \} \subseteq R^3$

$\Rightarrow$ Composition of behaviours :

$M : m \to n$ & $N : n \to p$

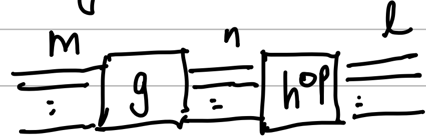
$B_1 = \{ (x, Mx) \mid x \in R^m \}$ $\quad B_2 = \{ (y, Ny) \mid x \in R^n \}$

$B_1 ; B_2 = \{ (x, z) \mid$ there exists $y \in R^n$ s.t. $(x, y) \in B_1$ and $(y, z) \in B_2 \}$

Ex 5.82

$g : m \to n \quad h : l \to n \Rightarrow h^{op} : n \to l$

we can compose $g ; h^{op}$



$B(g) = \{ (x, y) \subseteq R^m \times R^n \mid y = S(g)(x) \}$

$B(h^{op}) = \{ (y, z) \subseteq R^n \times R^l \mid z = S(h^{op})(y)$

$\cong y = S(h)(z) \}$

$B(g ; h^{op}) = \{ (x, z) \subseteq R^{m \times l} \mid y = S(g)(x) = S(h)(z) \}$