The Chartered Institute for IT
BCSWomen Lovelace Colloquium

The national conference for undergraduate and MSc women in computing

Submission Form

- 250 word abstract
- Deadline Feb 6th
- Women and non-binary students
- Best poster awards (for each year and for MSc)

Example Abstracts

The 19th annual BCSWomen Lovelace Colloquium will be held on Wednesday 8th and Thursday 9th April 2026, in-person, hosted by University of Bath.

Introduction to Quantum Programming and Semantics

Lecture 7: Copying and deleting

Chris Heunen

University of Edinburgh

Overview

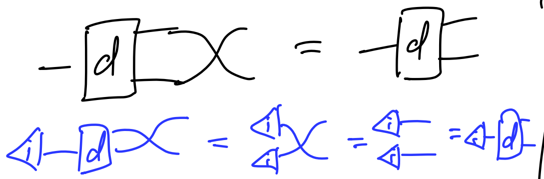
- Copying and deleting
- No-cloning and no-deleting
- Products
- Categories

Copying and deleting

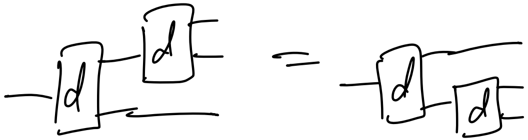
Copying machines

$$A \xrightarrow{d} A \otimes A$$
$$A \xrightarrow{e} \mathbb{C}^0 = \mathbb{C}$$

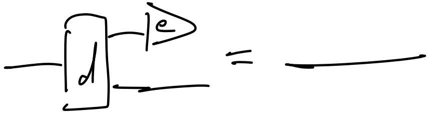
- commutative :

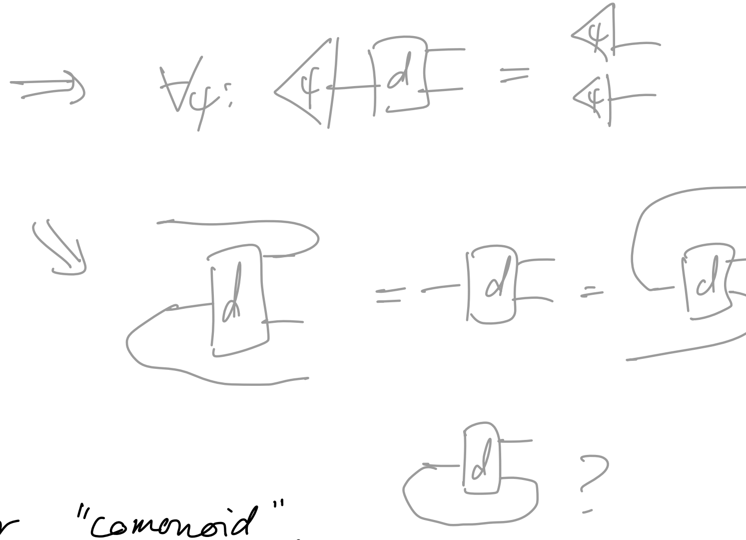


- associative :



- unital :





Such a process we'll call a "copying machine", or "comonoid",

e.g. $A = \mathbb{C}^n$

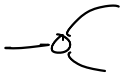
$d\,|i\rangle = |ii\rangle = |i\rangle \otimes |i\rangle$

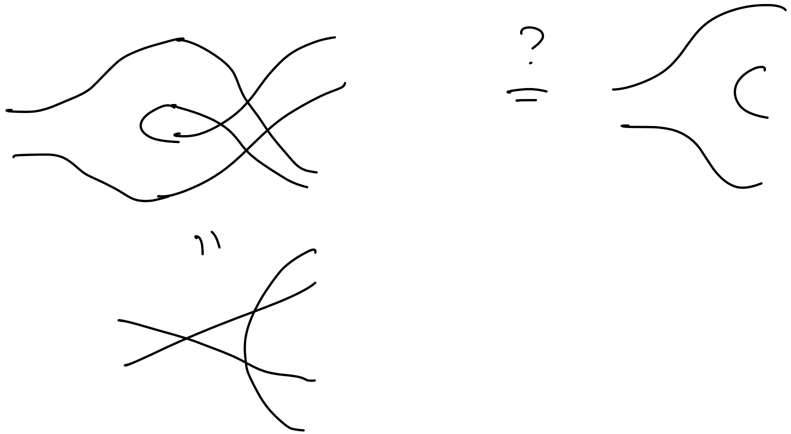
$e\,|i\rangle = 1$

$$d = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}$$

$$e = \begin{pmatrix} 1 & 1 \end{pmatrix}$$

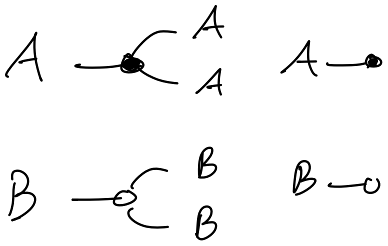
and we'll draw it as

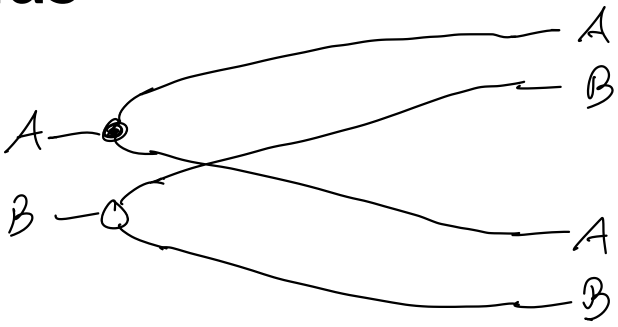
$d =$  $e =$ 

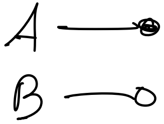

$\overset{?}{=}$
$=$

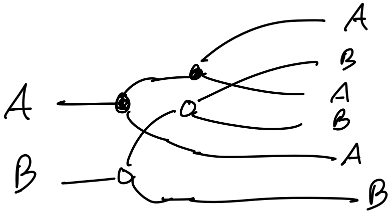
To distinguish different copying machines, we'll colour their pictures differently.

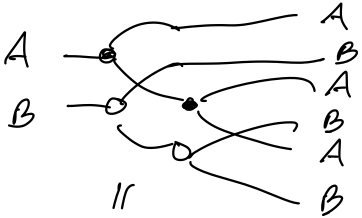
Combining comonoids

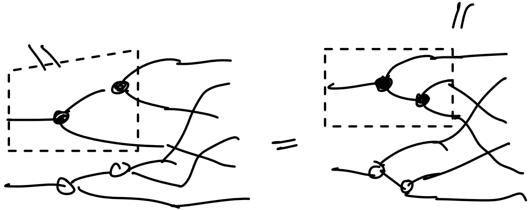




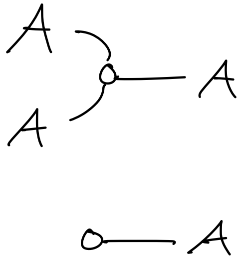




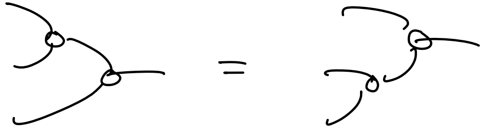




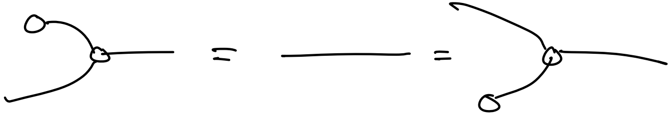
Monoids



s.t.

associative: 

$(m+n)+p = m+(n+p)$

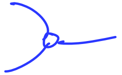
unital: 

$0+m = m = m+0$

(commutative: 

$n+m = m+n$)

e.g. $A = \mathbb{Z}$

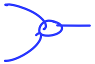


$= m+n$

 $= 0$

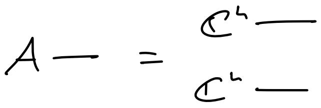
or

$A = M_n = \{n\text{-by-}n \text{ complex matrices}\}$

 = matrix multiplication

 $= \begin{pmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{pmatrix}$

Pairs of pants

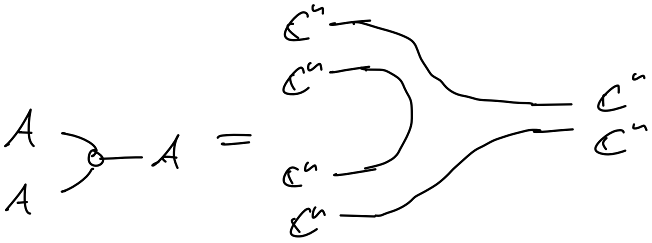
where 

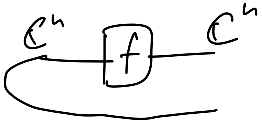
e.g. $A = \{n\text{-by-}n \text{ matrices}\}$

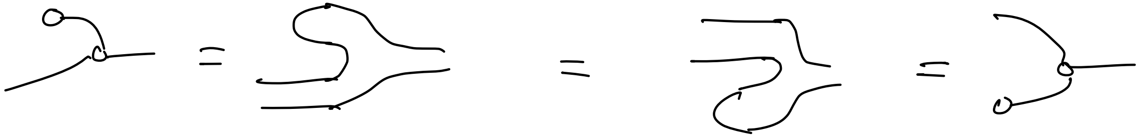
"$\cong \{f : \mathbb{C}^n \to \mathbb{C}^n \text{ linear}\}$"

$\cong (\mathbb{C}^n)^* \otimes \mathbb{C}^n \ni x$

$x = \sum_{i,j} x_{ij} \langle j| \otimes |i\rangle$



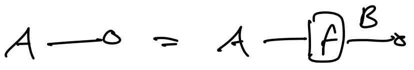


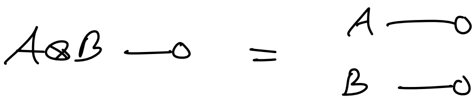
unital : 

No-cloning and no-deleting

Uniform deleting

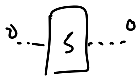
systematic deleting: for each type A, have 

s.t. for any process f: 

and: 

no-deleting theorem: if systematic deleting
and caps and cups ("entanglement") are available,
then there is a unique process 

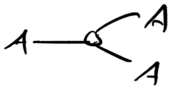
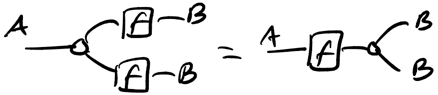
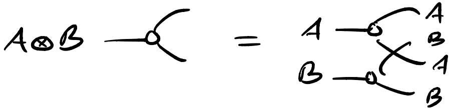
.

hence: unique scalar 

, so no quantities

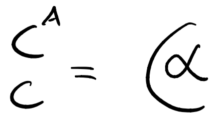
Uniform copying

Copying systematically means:

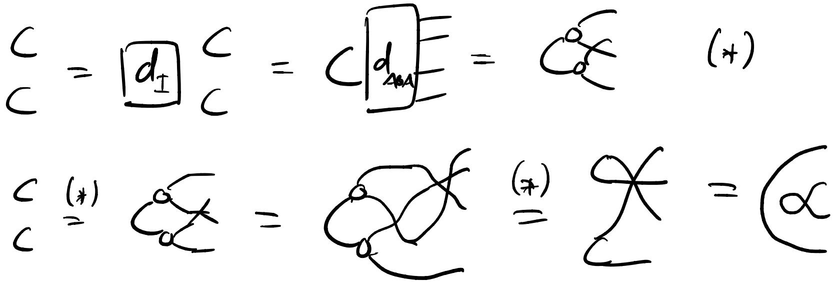
- for each type A, choose  s.t.:
- for each map f: 
- 

No-cloning theorem

Lem:

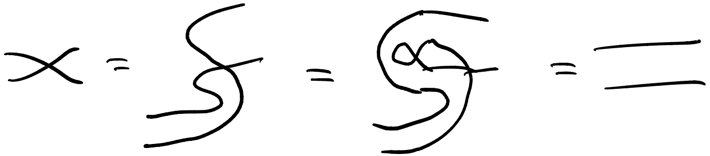


Pf:

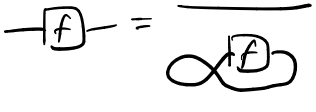


Lem:

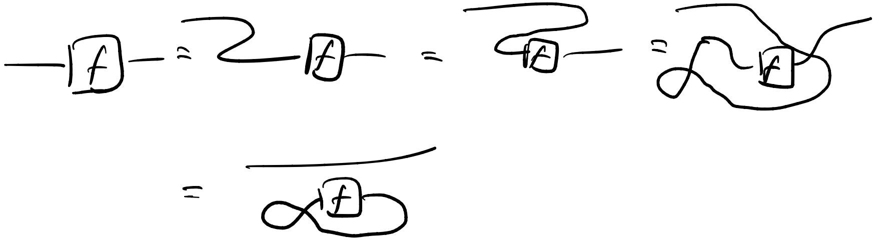
Pf:



Thm: If uniform copying, then



Pf:



Products

Universal property

$$\begin{tikzcd} & C \arrow[dl, "f"'] \arrow[dr, "g"] \arrow[d, dashed, "\exists!"] & \\ A & A \times B \arrow[l, "p_A"] \arrow[r, "p_B"'] & B \end{tikzcd}$$

$$c \mapsto (f(c), g(c))$$

Characterising products

<u>Thm</u>: In a monoidal category, $A \otimes B$ is a (categorical) product for each A and B if and only if

- there is uniform copying : $A \longrightarrow A \otimes A$
- there is uniform deleting : $A \longrightarrow I$
- 

Summary:

- Comonoids are copying machines
- No-cloning theorem holds syntactically
- Can recognise when semantics is classical via products
- Can interpret string diagrams generally in monoidal categories