



## Computabilidad y Complejidad

### **Práctica 3: Computación con membranas. Sistemas P**

# Computación con membranas. Sistemas P

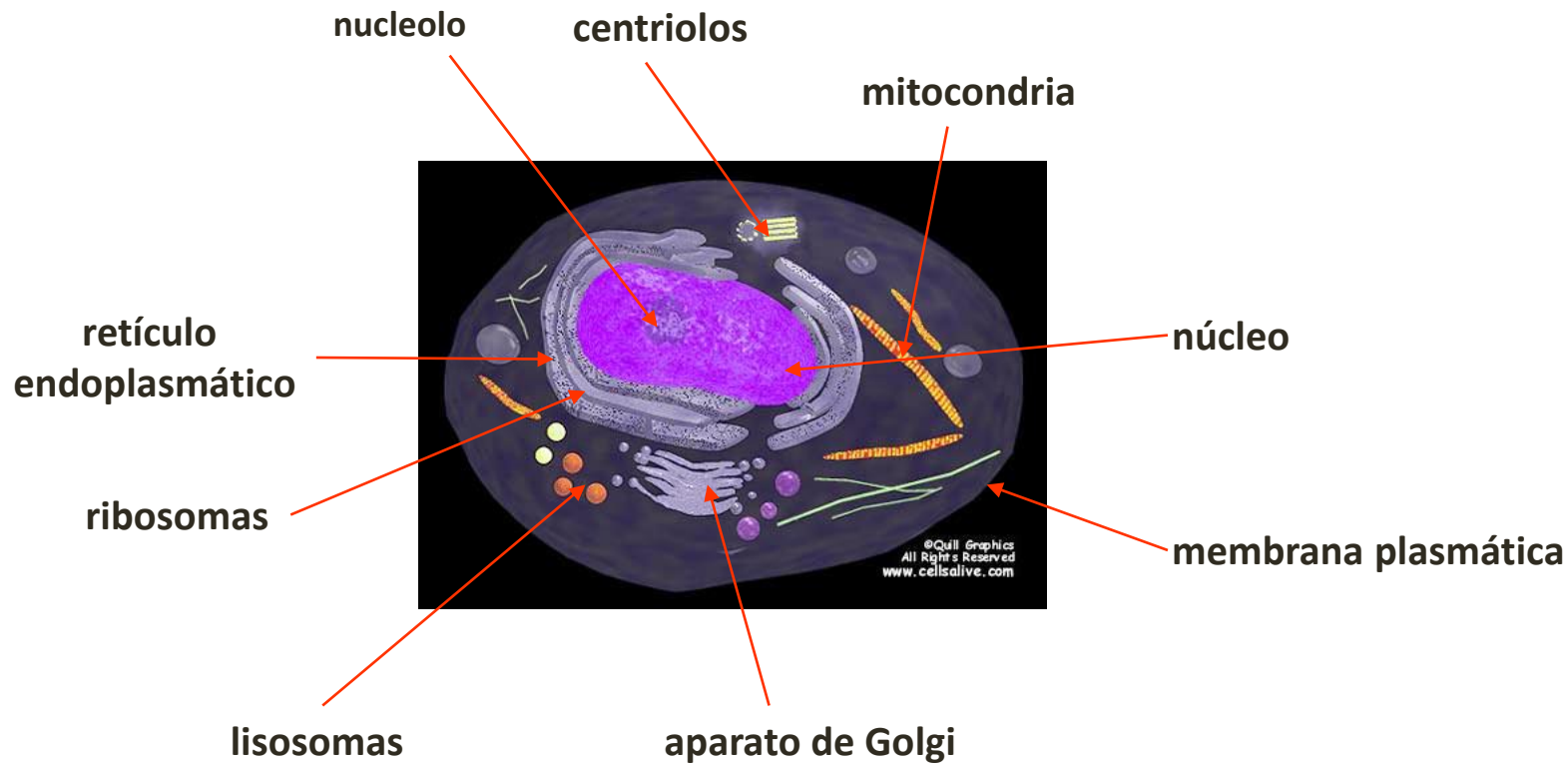
## Índice:

1. Conceptos básicos de biología celular.
2. Multiconjuntos.
3. Proyecciones de Parikh y conjuntos naturales.
4. Sistemas P. Sistemas P de transición.
5. Lenguajes y conjuntos.
6. Algunas clases definidas por los Sistemas P.
7. Actividades propuestas

## Bibliografía Básica

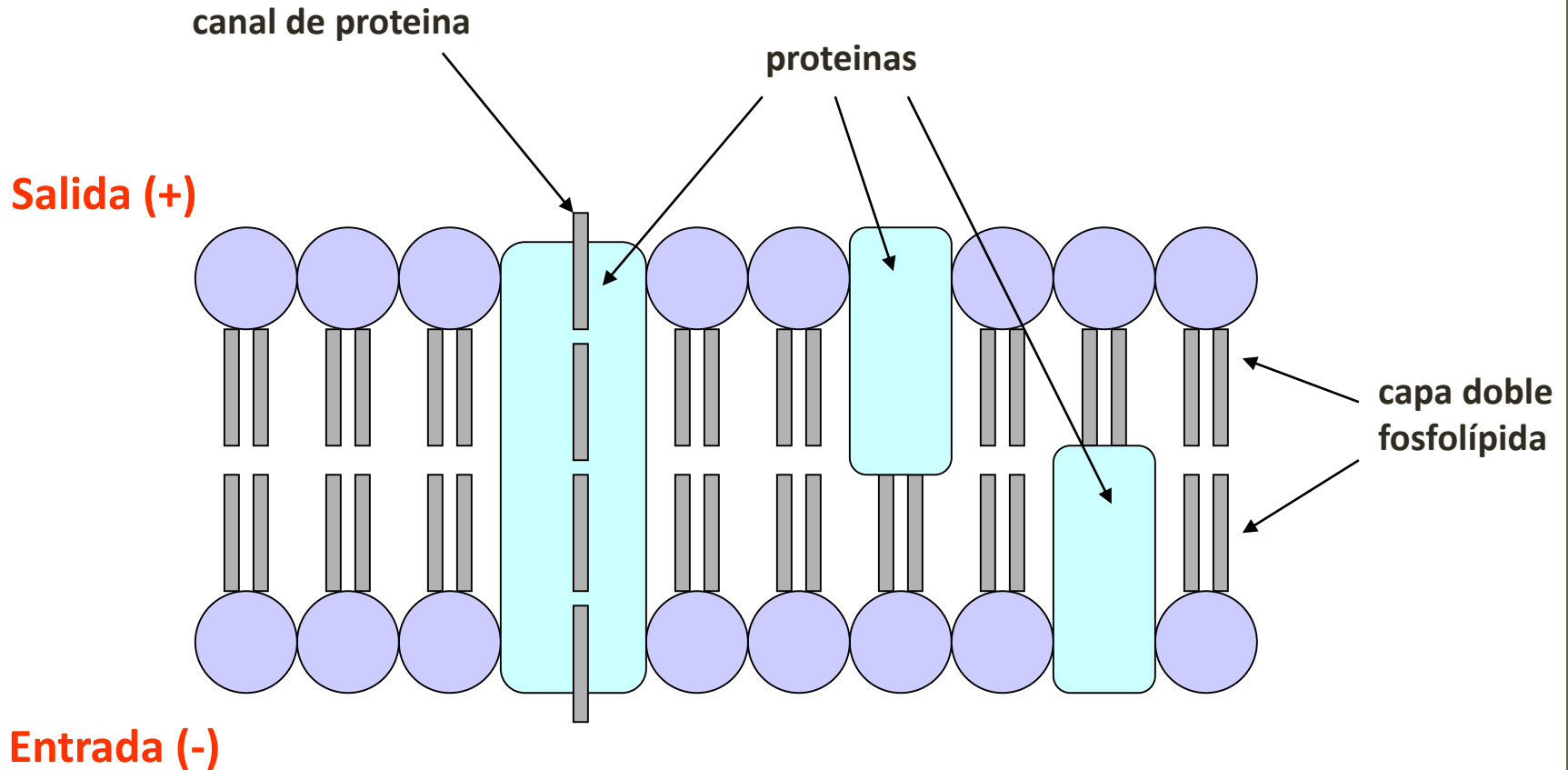
- Conceptos de genética. W. Clug, M. Cummings. Prentice Hall (5a. ed.). 1999.
- Membrane computing. An introduction. Gh. Păun. Springer. 2002.
- Computing with Bio-Molecules. Gh. Păun, (editor). Springer. 1998.
- The P systems web page (<http://ppage.psystems.eu/>)

## Estructura de la célula animal eucariota



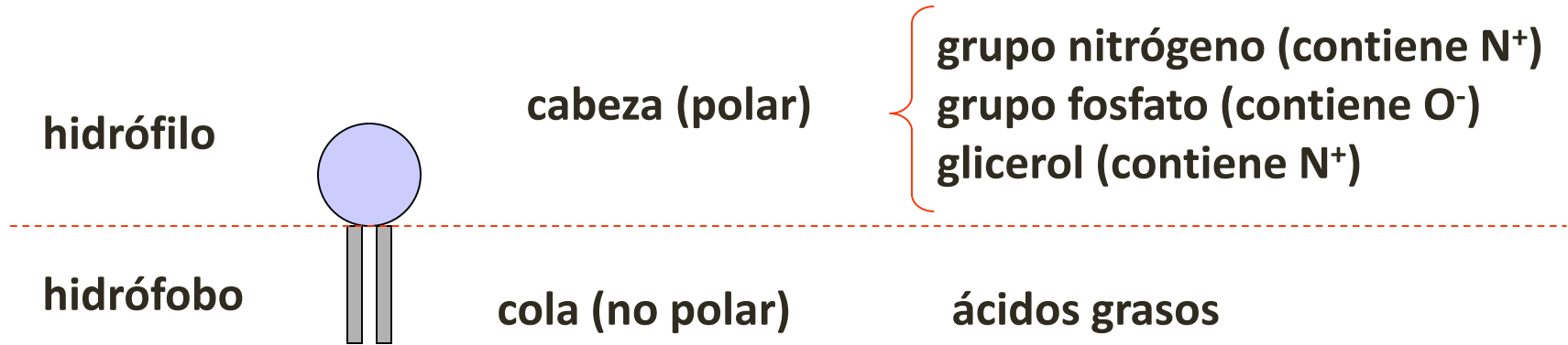
La célula viva puede considerarse un mecanismo real de procesamiento de información codificada bioquímicamente.

## Estructura de la membrana plasmática



Las membranas separan regiones en la célula viva. En cada región y de forma simultánea pueden producirse procesos bioquímicos que generan nuevas macromoléculas (típicamente proteínas). Los compuestos bioquímicos pueden migrar de una región a otra (incluso fuera de la célula) a través de las membranas.

## Estructura de la membrana fosfolípida



## Transporte de material a través de las membranas

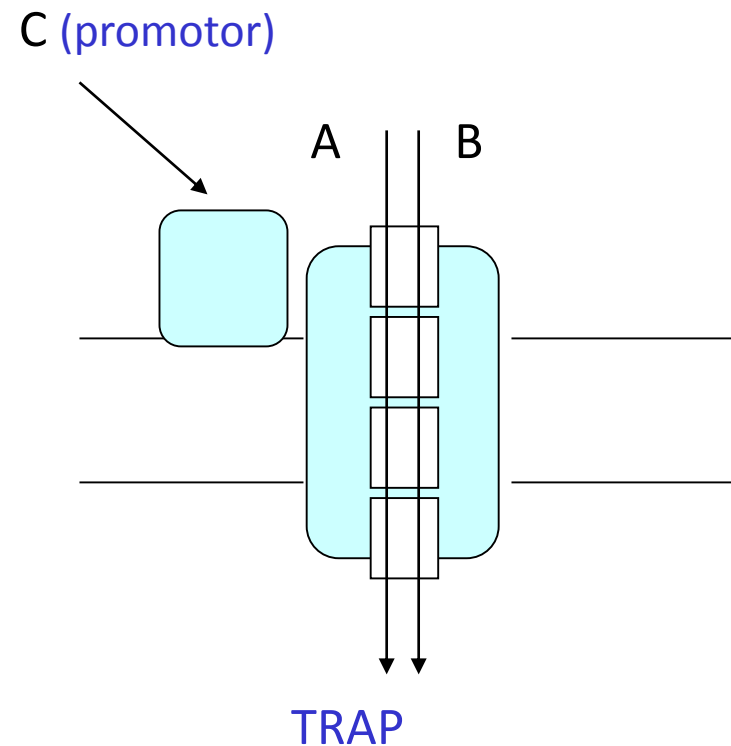
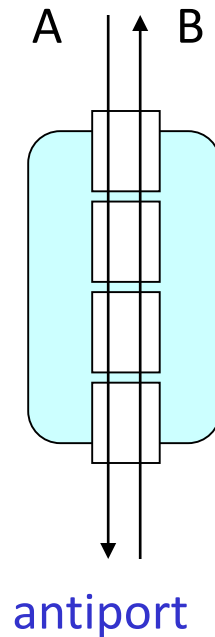
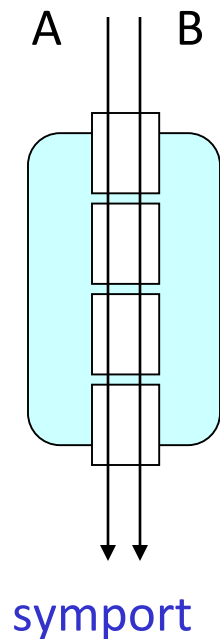
Transporte pasivo (sin consumo de energía)

Transporte activo (con consumo de energía, ATP)

Transporte mediante vesículas

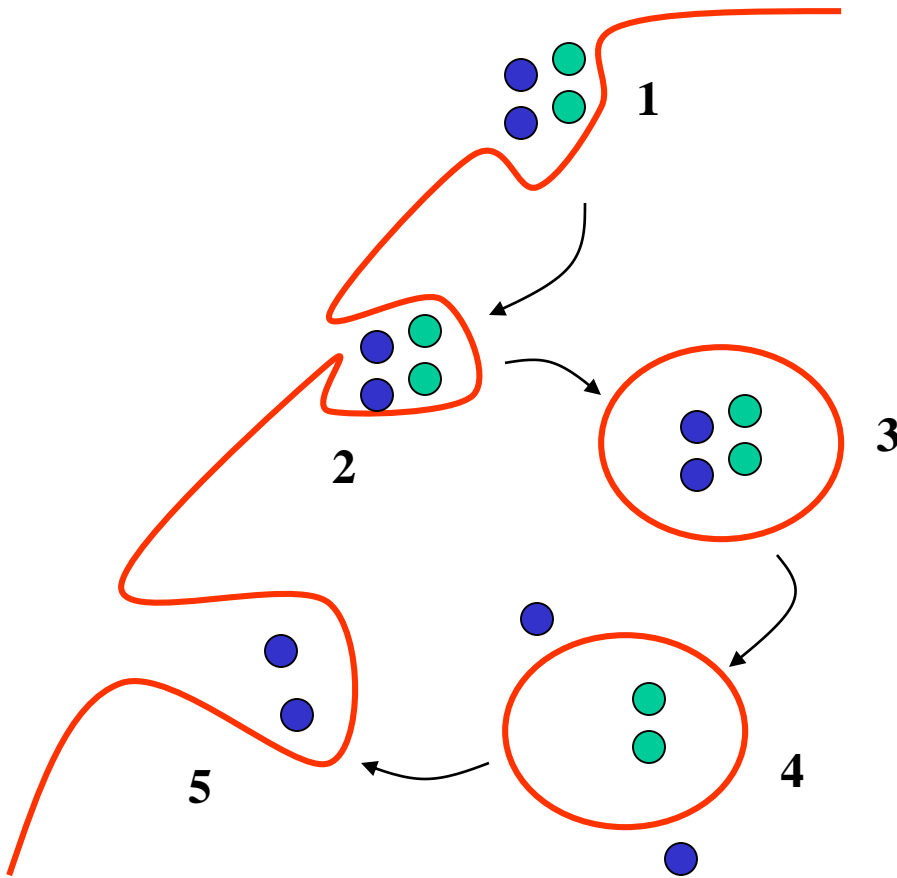
## Transporte activo

- Se realiza contra gradiente de la concentración
- Habitualmente se utilizan proteínas:
  - Mediante canales de proteínas
  - Mediante proteínas de acarreo (*carrier proteins*)
- Si intervienen más de una molécula en el transporte se dan diversas combinaciones



## Transporte activo mediante vesículas

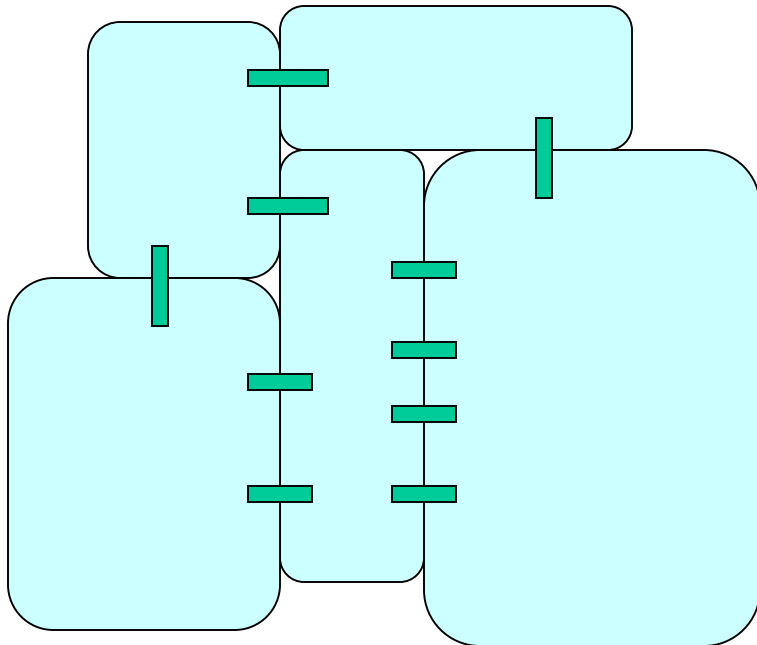
- Exocitosis (expulsión de una región)
- Endocitosis (introducción en una región). Ej. fagocitosis, pinocitosis



## Otras funciones de las proteínas de las membranas

- Actividad catalítica (enzimas)
- Reconocimiento de moléculas
- Enlace de moléculas

## Estructuras multicelulares: tejidos



Conexiones entre células :

1. Juntas de adhesión (*adhesion junctions*)
2. Juntas fuertes (*tight junctions*)
3. Juntas de huecos (*gap junctions*)



# División celular: Mitosis

Fase 1: Interfase

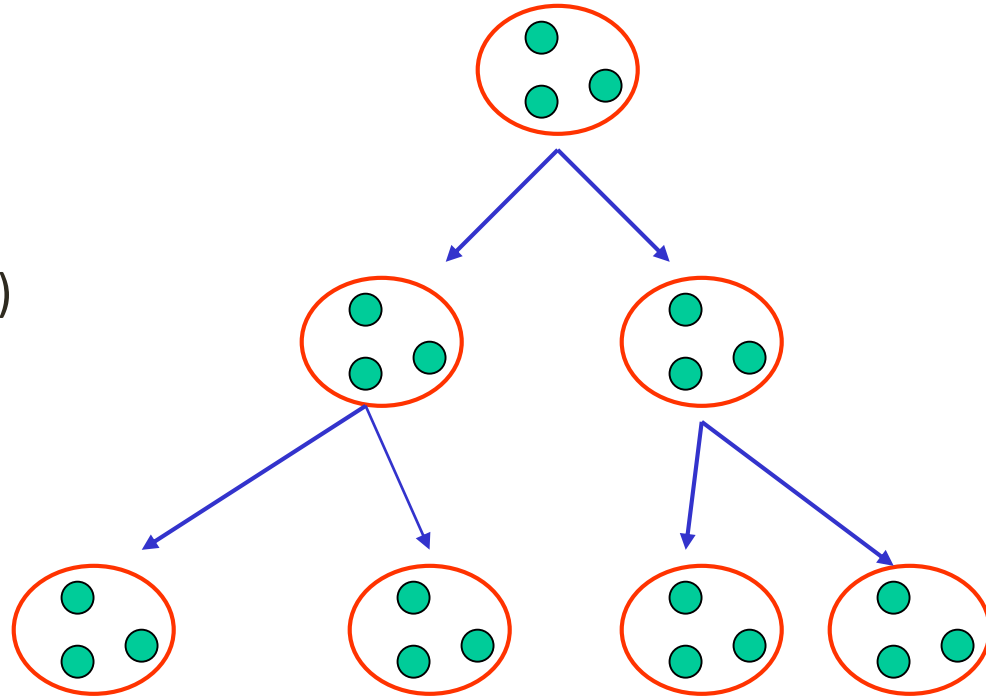
Fase 2: Profase

Fase 3: Prometafase

Fase 4: Metafase

Fase 5: Anafase

Fase 6: Telofase (citocinesis)



## Multiconjuntos

Un multiconjunto es un conjunto con multiplicidad asociada a sus elementos.

Dado el conjunto  $V$  un multiconjunto  $M$  sobre  $V$  se define mediante  $M: V \rightarrow \mathbb{N}$

- $M(a)$  es la **multiplicidad** del elemento  $a$
- $\text{supp}(M) = \{ a \in V : M(a) > 0 \}$  es **el soporte de  $M$** .
- $M_1 \subseteq M_2$  si  $\forall a \in V M_1(a) \leq M_2(a)$
- $(M_1 \cup M_2)(a) = M_1(a) + M_2(a)$
- $(M_1 - M_2)(a) = M_1(a) - M_2(a)$  (sólo se define si  $M_2 \subseteq M_1$  )
- La **amplificación de  $M$**  por  $n$  ( $\geq 0$ ) es  $(n \otimes M)(a) = n M(a) \forall a \in V$
- Dado el multiconjunto  $M$  y dados  $Y \subseteq X$  se define la **proyección de  $M$  sobre  $Y$**  como

$$pr_Y M(a) = \begin{cases} M(a) & \text{si } a \in Y \\ 0 & \text{en otro caso} \end{cases}$$

Todo multiconjunto con soporte finito admite una representación mediante la cadena  $a_1^{M(a_1)} a_2^{M(a_2)} \dots a_n^{M(a_n)}$  o cualquiera de sus permutaciones

## Proyecciones de Parikh y conjuntos naturales

Dado un alfabeto  $\Sigma = \{a_1, a_2, \dots, a_n\}$  y una cadena  $x = x_1 x_2 \dots x_m \in \Sigma^*$  se define la proyección de Parikh de  $x$  en  $\Sigma$  como la aplicación  $\Psi_\Sigma: \Sigma^* \rightarrow \mathbb{N}^n$

$$\Psi_\Sigma(x_1 x_2 \dots x_m) = (|x|_{a_1}, |x|_{a_2}, \dots, |x|_{a_n})$$

Dado un lenguaje  $L \subseteq \Sigma^*$   $\text{Ps}L = \{ \Psi_\Sigma(x) : x \in L \}$

Dada una familia de lenguajes FL **PsFL es la familia de conjuntos de proyecciones de Parikh** de los lenguajes  $L$  pertenecientes a FL

## Conjuntos naturales

Dado un alfabeto  $\Sigma = \{a_1, a_2, \dots, a_n\}$  y un lenguaje  $L \subseteq \Sigma^*$  se define el conjunto de naturales de  $L$  como  $NL = \{ n \in \mathbb{N} : x \in L \wedge |x| = n \}$ . Podemos definir el conjunto  $NL$  a partir de  $\text{Ps}L$  (como suma de las componentes de los vectores)

Dada una familia de lenguajes FL, **NFL es el conjunto de los conjuntos de naturales** definidos por lenguajes de FL.

$$\text{NFIN} \subset \text{NREG} = \text{NLIN} = \text{NCF} \subset \text{NCS} \subset \text{NRE}$$

## Computación con membranas. Sistemas P

Los sistemas P o modelos de computación con membranas fueron introducidos por Gh. Paun en 1998. Se inspiran en el funcionamiento de la célula eucariota animal y es un modelo de computación paralelo, distribuido y no determinista. Los principales ingredientes que se incorporan en el modelo son:

- Compartimentación en regiones separadas por membranas (cada región es un espacio de trabajo en el que operan reglas predefinidas)
- La información se representa mediante símbolos o cadenas, así como estructuras más complejas.
- La computación se basa en sistemas de reescritura (se asimila a las reacciones que suceden en la célula a nivel bioquímico)
- Existe la posibilidad de comunicar la información entre las regiones y de crear y destruir regiones dentro del modelo.

Los sistemas P se pueden considerar modelos de computación no convencionales. Forman parte de la computación bio-inspirada y son un paradigma de computación celular. Otros modelos celulares son las redes neuronales y las redes de procesadores bio-inspirados (además de los modelos clásicos de autómatas celulares).

## Algunos tipos de sistemas P

- Sistemas P de transición
- Sistemas P en tejidos
- Sistemas P catalíticos
- Sistemas P con comunicación
- Autómatas P
- Sistemas P con cadenas
- Sistemas P en splicing
- Sistemas P numéricos (*conformon P systems*)
- Sistemas P metabólicos
- Sistemas P con membranas activas y objetos en las membranas
- Sistemas P con “impulsos neuronales” (*spiking neuron P systems*)
- Sistemas P estocásticos y probabilistas
- Sistemas P en colonias
- etc, etc.

En esta práctica nos centraremos sólo en los sistemas P de transición que fueron los primeros propuestos por Paun. Además, permiten la universalidad con estructuras de membranas muy simples.

## Sistemas P de transición

Un sistema P de transición de grado  $m$  se define como la tupla

$$\Pi = ( O, \mu, w_1, \dots, w_m, (R_1, \rho_1), \dots, (R_m, \rho_m), i_0 )$$

$O$  alfabeto de objetos<sup>1</sup>

$\mu$  estructura de membranas (representable mediante un árbol)

$w_1, \dots, w_m$  cadenas asociadas a cada membrana (multiconjuntos)

$(R_1, \rho_1), \dots, (R_m, \rho_m)$  reglas de evolución y relaciones de prioridad

$i_0 \in \{ 1, \dots, m \} \cup \{ \infty \}$  membrana de salida

<sup>1</sup> En algunas variantes de sistemas P de transición, el alfabeto  $O$  se divide en tres alfabetos: objetos, *catalizadores* y objetos de salida.

Nosotros en la presente práctica no haremos distinción entre los objetos.

## Sistemas P de transición

$$\Pi = ( O, \mu, w_1, \dots, w_m, (R_1, \rho_1), \dots, (R_m, \rho_m), i_0 )$$

$(R_1, \rho_1), \dots, (R_m, \rho_m)$  reglas de evolución y relaciones de prioridad

$R_i$  es un conjunto finito de reglas de evolución

$$(u, v) \quad \text{o} \quad u \rightarrow v$$

donde  $u \in O^*$  y  $v = v'$  o  $v = v'\delta$  con  $v' \in Coop^*$  y  $\delta \notin O$

$$Coop = \{ a_{here}, a_{out}, a_{inj} \mid a \in O, 1 \leq j \leq m \}$$

$inj_j$  denota direccionamiento por objetivo

El símbolo  $\delta$  denota la disolución de la membrana

Dada la regla  $u \rightarrow v$  llamaremos radio de la regla a  $|u|$

$\rho_i$  es un orden parcial sobre  $R_i$  (prioridad)

Un sistema  $\Pi$  es **cooperativo** si contiene alguna regla con radio mayor que 1

# Sistemas P de transición

*(representación mediante diagramas)*

$$\Pi = ( O, \mu, w_1, \dots, w_m, (R_1, \rho_1), \dots, (R_m, \rho_m), i_0 )$$

## Ejemplo

$$O = \{ a, b, d, e, f \} \quad i_0 = 1$$

$$\mu = [ {}_1 [ {}_2 [ {}_3 [ {}_3 ] {}_2 ] {}_1 ]$$

$$w_1 = \lambda \quad R_1 = \{ d \rightarrow d_{out} \}$$

$$w_2 = \lambda \quad R_2 = \{ b \rightarrow d, d \rightarrow de, r_1: ff \rightarrow f, r_2: f \rightarrow \delta \} \quad \rho_1 = \emptyset$$

$$w_3 = af \quad R_3 = \{ a \rightarrow ab, a \rightarrow b\delta, f \rightarrow ff \} \quad \rho_2 = \{ r_1 > r_2 \}$$

$$\rho_3 = \emptyset$$



1

2

$d \rightarrow d_{out}$

3

$b \rightarrow d$

$d \rightarrow de$

$(ff \rightarrow f) > (f \rightarrow \delta)$

$af$

$a \rightarrow ab$

$a \rightarrow b\delta$

$f \rightarrow ff$

## Configuraciones, transiciones, lenguajes y conjuntos de los sistemas P (I)

$$\Pi = (O, \mu, w_1, \dots, w_m, (R_1, \rho_1), \dots, (R_m, \rho_m), i_0)$$

$(\mu', w_{i1}, \dots, w_{ik})$  *configuración de  $\Pi$*

$(\mu, w_1, \dots, w_m)$  *configuración inicial de  $\Pi$*

### Transiciones

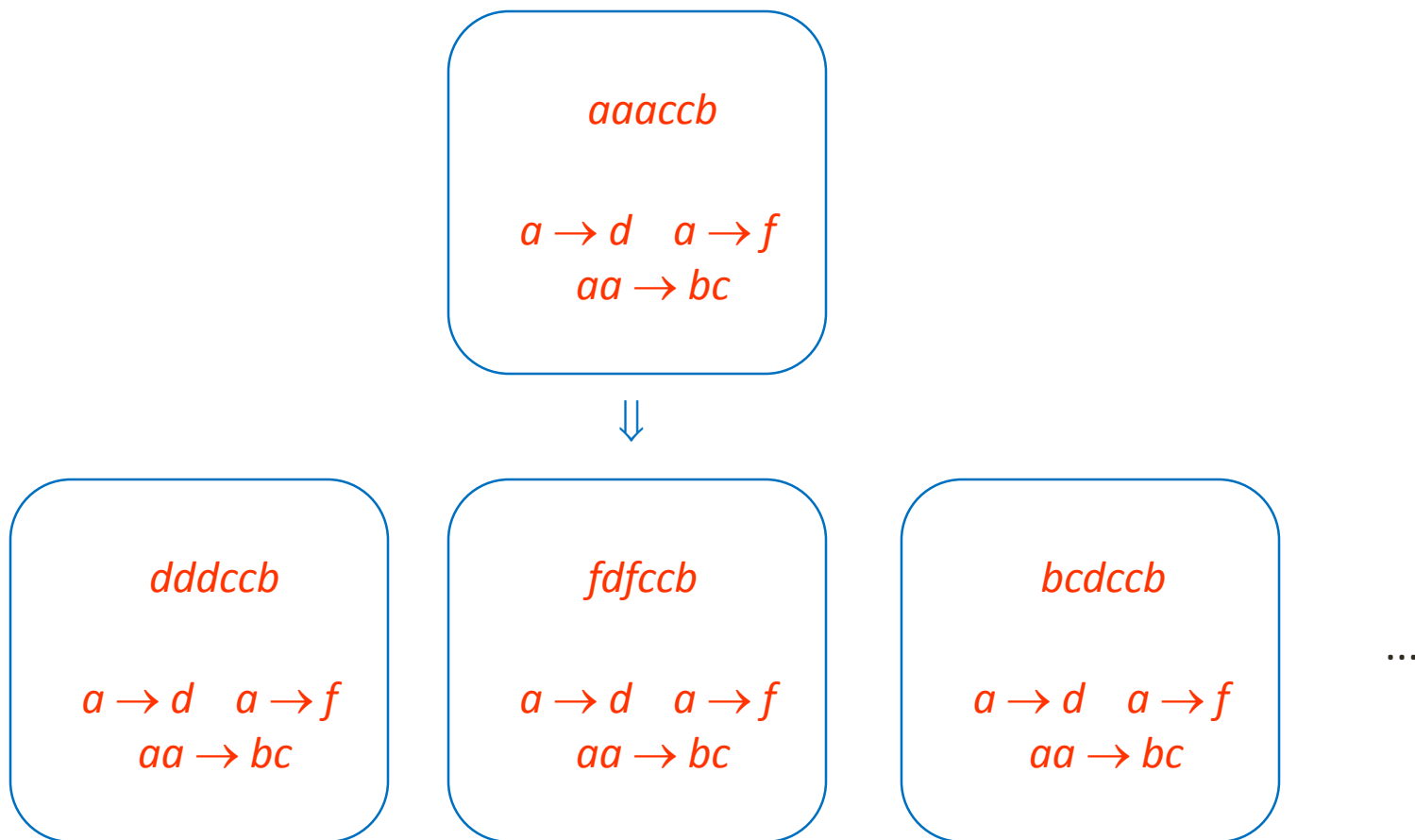
$$C_1 = (\mu', w_{i1}, \dots, w_{ik}) \quad C_2 = (\mu'', w'_{j1}, \dots, w'_{jl})$$

$$C_1 \Rightarrow C_2$$

# Configuraciones, transiciones, lenguajes y conjuntos de los sistemas P (II)

## *Aplicación de las reglas de evolución*

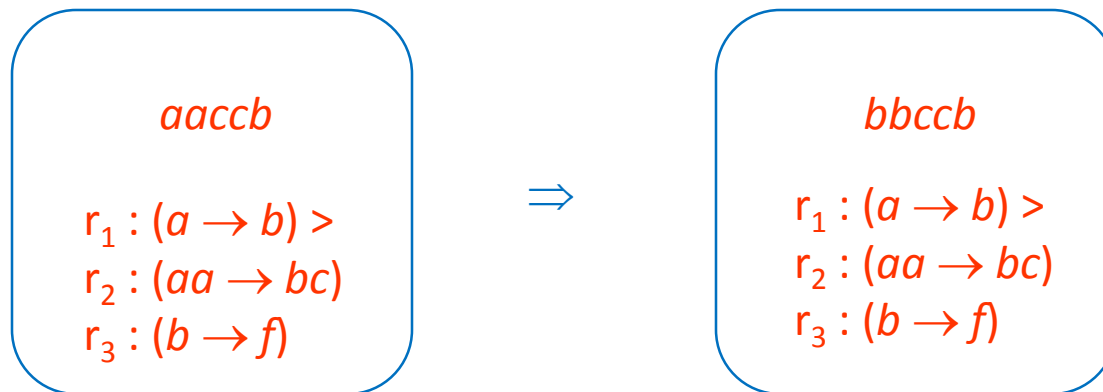
*Todas las reglas que se puedan aplicar en una región se aplican en paralelo*



## Configuraciones, transiciones, lenguajes y conjuntos de los sistemas P (III)

### *Aplicación de las reglas de evolución*

*La prioridad entre reglas puede impedir que una regla se aplique*

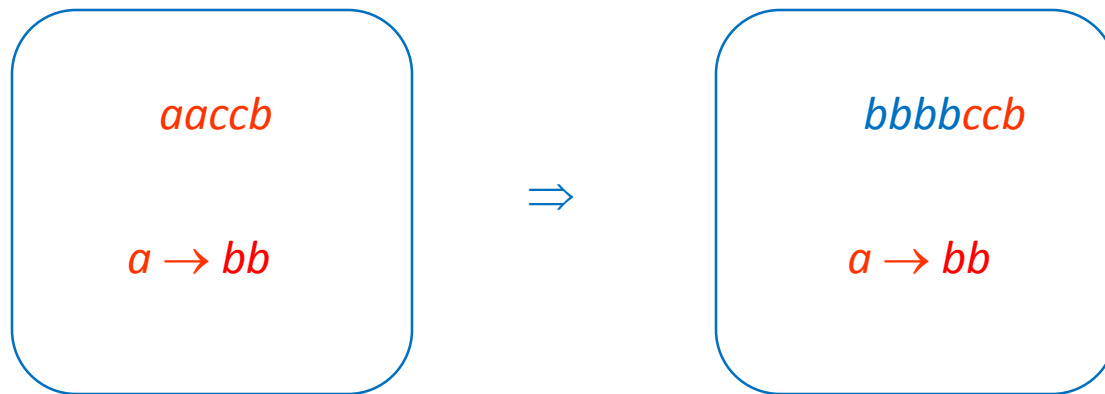


Una interpretación menos rígida permite que se apliquen reglas con menor prioridad si es posible y siempre que se hayan aplicado todas las reglas más prioritarias.  
(en nuestro ejemplo obtendríamos el multiconjunto **bbccf**)

# Configuraciones, transiciones, lenguajes y conjuntos de los sistemas P (IV)

## Aplicación de las reglas de evolución

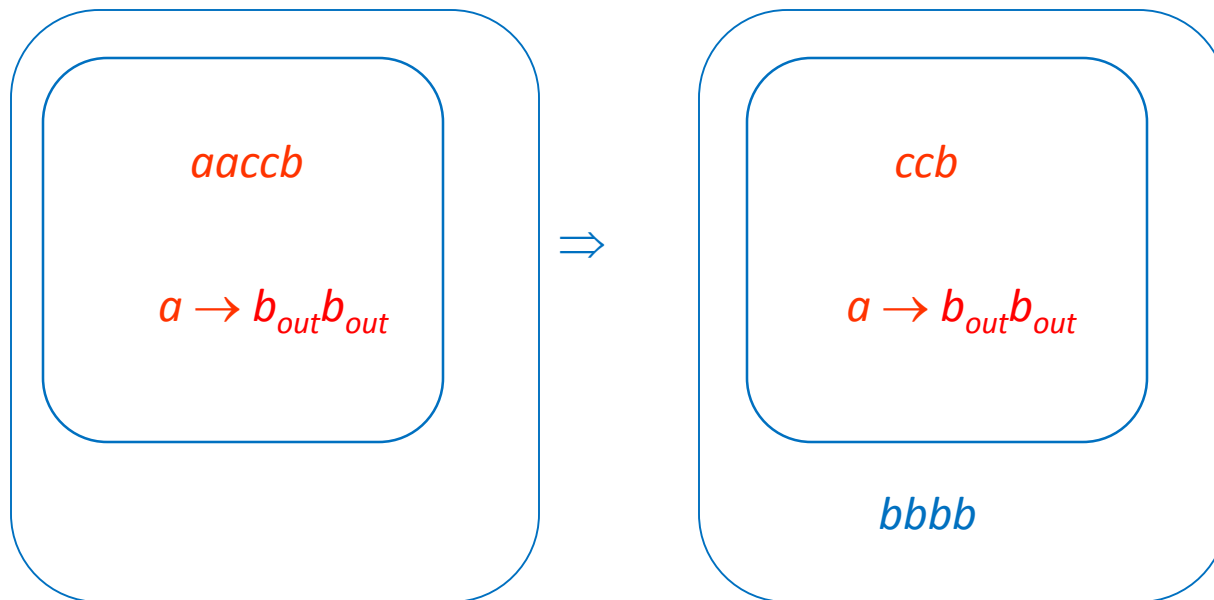
*Reglas con símbolos  $a_{\text{here}}$  hacen que los símbolos permanezcan en la región*



# Configuraciones, transiciones, lenguajes y conjuntos de los sistemas P (V)

## Aplicación de las reglas de evolución

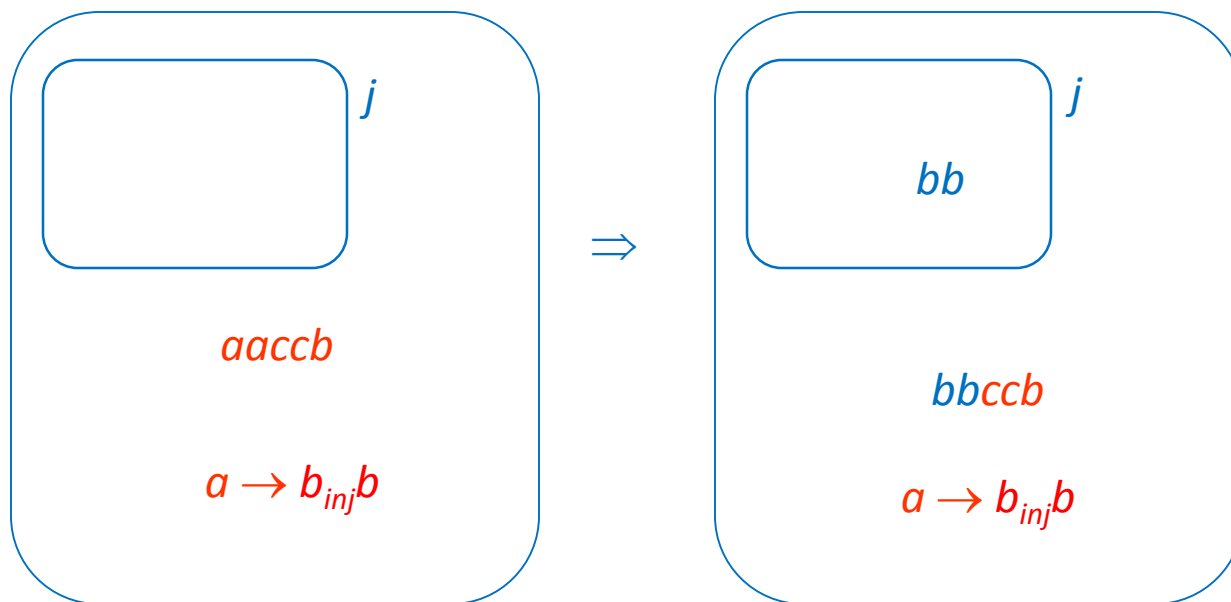
*Reglas con símbolos  $a_{out}$  hacen que los símbolos pasen a la región inmediatamente superior de acuerdo con  $\mu$  (o salgan del sistema )*



# Configuraciones, transiciones, lenguajes y conjuntos de los sistemas P (VI)

## Aplicación de las reglas de evolución

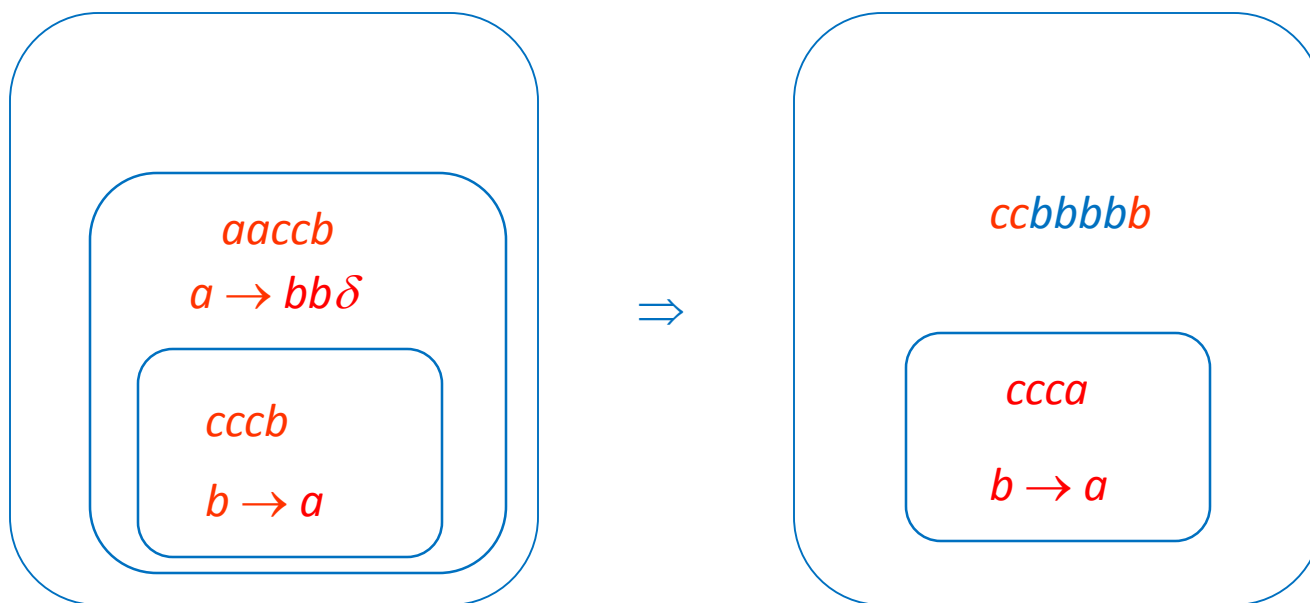
*Reglas con símbolos  $a_{inj}$  hacen que los símbolos pasen a la región  $j$  siempre que  $j$  sea una región adyacente inferior de acuerdo con  $\mu$*



## Configuraciones, transiciones, lenguajes y conjuntos de los sistemas P (VII)

### Aplicación de las reglas de evolución

*Reglas con el símbolo  $\delta$  hacen que, tras la aplicación de la regla, la membrana correspondiente desaparezca y los objetos internos se hereden en la membrana inmediatamente superior de acuerdo con  $\mu$*





## Configuraciones, transiciones, lenguajes y conjuntos de los sistemas P (VIII)

### *Conjunto generado en modo interno*

$$\Pi = ( O, \mu, w_1, \dots, w_m, (R_1, \rho_1), \dots, (R_m, \rho_m), i_0 )$$

$\Psi_O(w)$  (Proyección de Parikh asociada a O)

$Ps(\Pi)$  (Conjunto de Parikh)  $N(\Pi) \equiv Ps(\Pi)$

(en este caso no haremos distinción entre las proyecciones de Parikh y los conjuntos de naturales que inducen)

### *Lenguaje generado en modo externo*

$$\Pi = ( O, \mu, w_1, \dots, w_m, (R_1, \rho_1), \dots, (R_m, \rho_m), \infty )$$

$L(\Pi)$  Cadenas recogidas desde el exterior

(incluyendo permutaciones cuando se expulsan varios objetos)

## Algunas clases definidas por los sistemas P

Denotaremos por  $NOP_m(\alpha, tar)$  a la familia de conjuntos de naturales  $N(\Pi)$  siendo  $\Pi$  un sistema P de grado máximo  $m \geq 1$  con reglas de tipo  $\alpha$  y direccionamiento por objetivo.  $\alpha \in \{coo, ncoo, cat^1\}$  (si  $m=*$  el grado del sistema no está acotado en grado)

$$\underline{\text{Lema}} \quad NOP_*(\alpha, tar) = NOP_m(\alpha, tar) \quad m \geq 2$$

$$\underline{\text{Lema}} \quad NOP_m(ncoo, tar) \subseteq NOP_m(cat, tar) \subseteq NOP_m(coo, tar) \quad m \geq 1$$

$$\underline{\text{Lema}} \quad NOP_*(ncoo, tar) \subseteq NOP_*(cat, tar) \subseteq NOP_*(coo, tar)$$

$$\underline{\text{Lema}} \quad NOP_*(coo, tar) = NOP_m(coo, tar) = NRE \quad m \geq 1$$

$$\underline{\text{Lema}} \quad NOP_*(ncoo, tar) = NOP_1(ncoo) = NCF$$

<sup>1</sup> cat denota que el sistema utiliza catalizadores que son objetos que no se transforman en ninguna regla

# Algunas clases definidas por los sistemas P

*Denotaremos por  $NOP_m(\alpha, tar, \delta)$  a la familia de conjuntos de naturales  $N(\Pi)$  siendo  $\Pi$  un sistema P de grado máximo  $m \geq 1$  con reglas de tipo  $\alpha$ , direccionamiento por objetivo y disolución de membranas  $\alpha \in \{coo, ncoo, cat\}$*

**Lema**  $NOP_*(ncoo, tar) \subset NOP_2(ncoo, tar, \delta)$

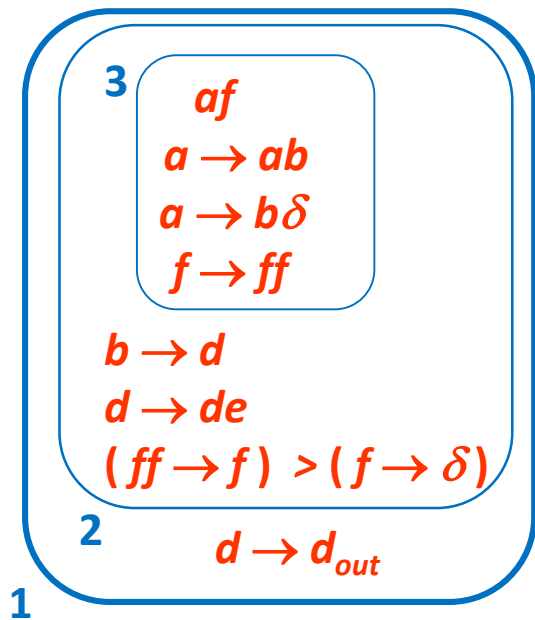
*Denotaremos por  $NOP_m(\alpha, tar, pri)$  a la familia de conjuntos de naturales  $N(\Pi)$  siendo  $\Pi$  un sistema P de grado máximo  $m \geq 1$  con reglas de tipo  $\alpha$ , direccionamiento por objetivo y prioridades  $\alpha \in \{coo, ncoo, cat\}$*

**Lema.**  $NOP_2(cat, tar, pri) = NRE$

# Actividades propuestas

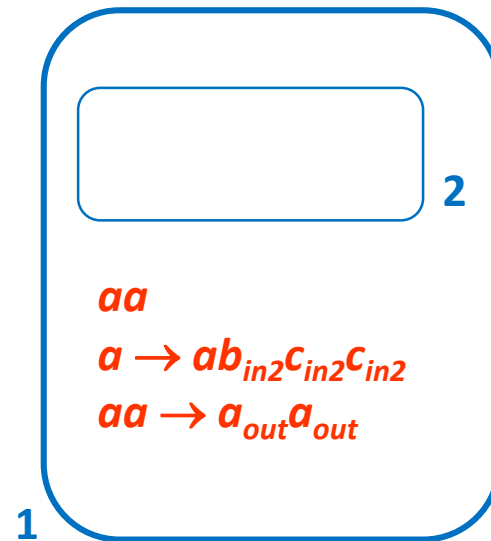
1. Especifique el conjunto de naturales calculado por cada uno de los siguientes sistemas P

(a)



$$i_0 = 1$$

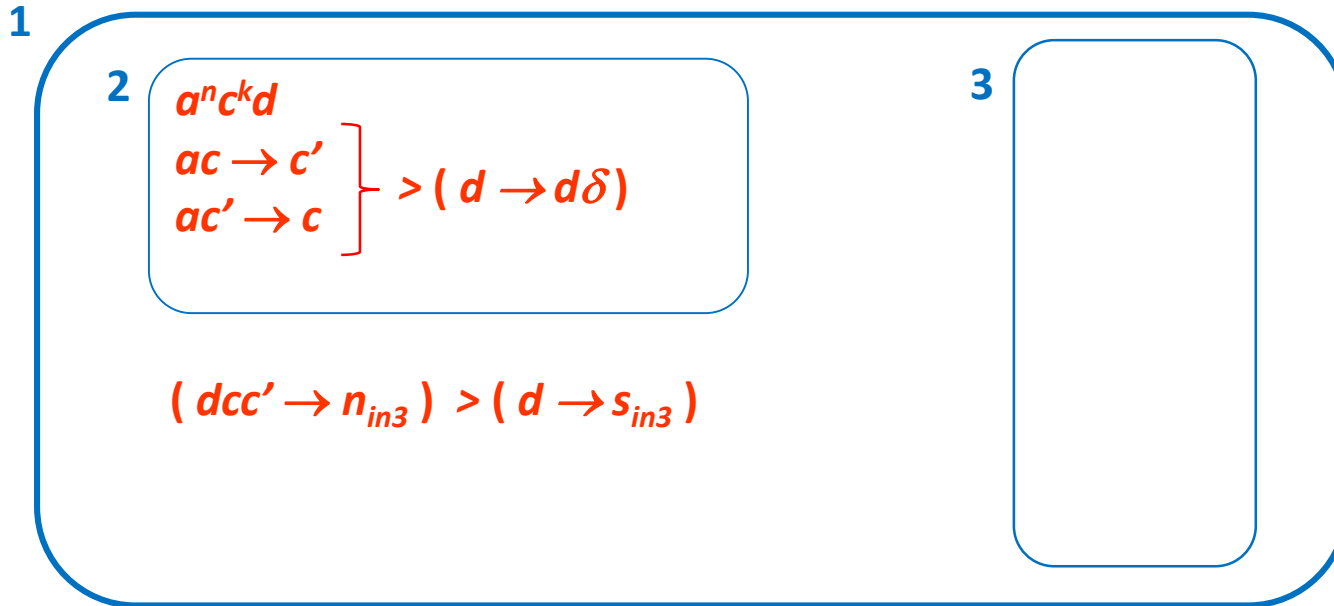
(b)



$$i_0 = 2$$

## Actividades propuestas

2. Dado el siguiente sistema P, establezca cuándo el sistema calcula como salida “s” y cuándo calcula como salida “n” (considere la región número 3 como la de salida)



$$i_0 = 3$$

3. Diseñe un módulo Mathematica que, dado como entrada un valor entero n, proporcione como salida la configuración del sistema P del ejercicio 1(b) después de aplicar n transiciones