

Tema 3-4

Recurrencias y fórmulas explícitas



UNIVERSIDAD
DE CHILE

Arturo Merino



Recuerdo/Plan

- **Recuerdo:**

- **Recuerdo:**
 - Principio biyectivo y demostraciones combinatoriales

- **Recuerdo:**
 - Principio biyectivo y demostraciones combinatoriales
 - Principio del palomar

- **Recuerdo:**

- Principio biyectivo y demostraciones combinatoriales
- Principio del palomar

- **Plan:**

- **Recuerdo:**

- Principio biyectivo y demostraciones combinatoriales
- Principio del palomar

- **Plan:**

- Recurrencias

- **Recuerdo:**

- Principio biyectivo y demostraciones combinatoriales
- Principio del palomar

- **Plan:**

- Recurrencias
- Formas elementales de resolución

Recurrencias

Recurrencias

- Hasta ahora: conteo via *principios*.

Recurrencias

- Hasta ahora: conteo via *principios*.

Def.

Una *(relación de) recurrencia* para una sucesión $(a_n)_{n \geq 0}$ es una relación que expresa a_n en términos de a_k para $k < n$.

Recurrencias

- Hasta ahora: conteo via *principios*.

Def.

Una *(relación de) recurrencia* para una sucesión $(a_n)_{n \geq 0}$ es una relación que expresa a_n en términos de a_k para $k < n$.

Muchas veces la recurrencia vendrá acompañada de *condiciones iniciales*, que son valores específicos de a_k para algunos $k < n$.

Recurrencias

- Hasta ahora: conteo via *principios*.

Def.

Una *(relación de) recurrencia* para una sucesión $(a_n)_{n \geq 0}$ es una relación que expresa a_n en términos de a_k para $k < n$.

Muchas veces la recurrencia vendrá acompañada de *condiciones iniciales*, que son valores específicos de a_k para algunos $k < n$.

- **Ejemplos:**

Recurrencias

- Hasta ahora: conteo via *principios*.

Def.

Una *(relación de) recurrencia* para una sucesión $(a_n)_{n \geq 0}$ es una relación que expresa a_n en términos de a_k para $k < n$.

Muchas veces la recurrencia vendrá acompañada de *condiciones iniciales*, que son valores específicos de a_k para algunos $k < n$.

- **Ejemplos:**

- $f_n = f_{n-1} + f_{n-2}$ para $n \geq 2$. $f_0 = 0$, $f_1 = 1$.

Recurrencias

- Hasta ahora: conteo via *principios*.

Def.

Una *(relación de) recurrencia* para una sucesión $(a_n)_{n \geq 0}$ es una relación que expresa a_n en términos de a_k para $k < n$.

Muchas veces la recurrencia vendrá acompañada de *condiciones iniciales*, que son valores específicos de a_k para algunos $k < n$.

• Ejemplos:

- $f_n = f_{n-1} + f_{n-2}$ para $n \geq 2$. $f_0 = 0$, $f_1 = 1$.
- $x_n = \cos(x_{\lceil \frac{n}{2} \rceil}) + 2^{x_{n-1}}$ para $n \geq 3$. $x_2 = x_1 = x_0 = \pi$.

Recurrencias

- Hasta ahora: conteo via *principios*.

Def.

Una *(relación de) recurrencia* para una sucesión $(a_n)_{n \geq 0}$ es una relación que expresa a_n en términos de a_k para $k < n$.

Muchas veces la recurrencia vendrá acompañada de *condiciones iniciales*, que son valores específicos de a_k para algunos $k < n$.

• Ejemplos:

- $f_n = f_{n-1} + f_{n-2}$ para $n \geq 2$. $f_0 = 0$, $f_1 = 1$.

- $x_n = \cos(x_{\lceil \frac{n}{2} \rceil}) + 2^{x_{n-1}}$ para $n \geq 3$. $x_2 = x_1 = x_0 = \pi$.

- Fórmula cerrada **X**

Recurrencias

- Hasta ahora: conteo via *principios*.

Def.

Una *(relación de) recurrencia* para una sucesión $(a_n)_{n \geq 0}$ es una relación que expresa a_n en términos de a_k para $k < n$.

Muchas veces la recurrencia vendrá acompañada de *condiciones iniciales*, que son valores específicos de a_k para algunos $k < n$.

• Ejemplos:

- $f_n = f_{n-1} + f_{n-2}$ para $n \geq 2$. $f_0 = 0$, $f_1 = 1$.

- $x_n = \cos(x_{\lceil \frac{n}{2} \rceil}) + 2^{x_{n-1}}$ para $n \geq 3$. $x_2 = x_1 = x_0 = \pi$.

- Fórmula cerrada ✗

- Sencillo y eficiente de computar ✓

Recurrencias

- Hasta ahora: conteo via *principios*.

Def.

Una *(relación de) recurrencia* para una sucesión $(a_n)_{n \geq 0}$ es una relación que expresa a_n en términos de a_k para $k < n$.

Muchas veces la recurrencia vendrá acompañada de *condiciones iniciales*, que son valores específicos de a_k para algunos $k < n$.

• Ejemplos:

- $f_n = f_{n-1} + f_{n-2}$ para $n \geq 2$. $f_0 = 0$, $f_1 = 1$.

- $x_n = \cos(x_{\lceil \frac{n}{2} \rceil}) + 2^{x_{n-1}}$ para $n \geq 3$. $x_2 = x_1 = x_0 = \pi$.

- Fórmula cerrada ✗
- Sencillo y eficiente de computar ✓
- Otra forma de entender el problema de conteo ✓

Recurrencias

- Hasta ahora: conteo via *principios*.

Def.

Una *(relación de) recurrencia* para una sucesión $(a_n)_{n \geq 0}$ es una relación que expresa a_n en términos de a_k para $k < n$.

Muchas veces la recurrencia vendrá acompañada de *condiciones iniciales*, que son valores específicos de a_k para algunos $k < n$.

• Ejemplos:

- $f_n = f_{n-1} + f_{n-2}$ para $n \geq 2$. $f_0 = 0$, $f_1 = 1$.

- $x_n = \cos(x_{\lceil \frac{n}{2} \rceil}) + 2^{x_{n-1}}$ para $n \geq 3$. $x_2 = x_1 = x_0 = \pi$.

- Fórmula cerrada ✗
- Sencillo y eficiente de computar ✓
- Otra forma de entender el problema de conteo ✓
- Primer paso para una fórmula cerrada ✓

Ejemplos

- $b_n \leftarrow$ Número de string binarios

Ejemplos

- $b_n \leftarrow$ Número de string binarios
- ¿Cómo obtener relaciones de recurrencia?

Ejemplos

- $b_n \leftarrow$ Número de string binarios
- ¿Cómo obtener relaciones de recurrencia?
- **Usualmente:** particionar por casos y reconocer casos más pequeños.

Ejemplos

- $b_n \leftarrow$ Número de string binarios
- ¿Cómo obtener relaciones de recurrencia?
- **Usualmente:** particionar por casos y reconocer casos más pequeños.
 - Principio aditivo.

Ejemplos

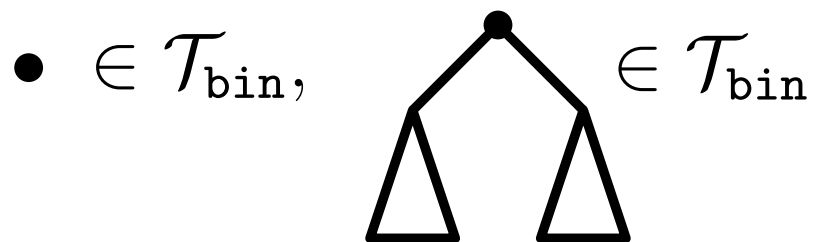
- $b_n \leftarrow$ Número de string binarios
- ¿Cómo obtener relaciones de recurrencia?
- **Usualmente:** particionar por casos y reconocer casos más pequeños.
 - Principio aditivo.
- **Ejemplos:**

Ejemplos

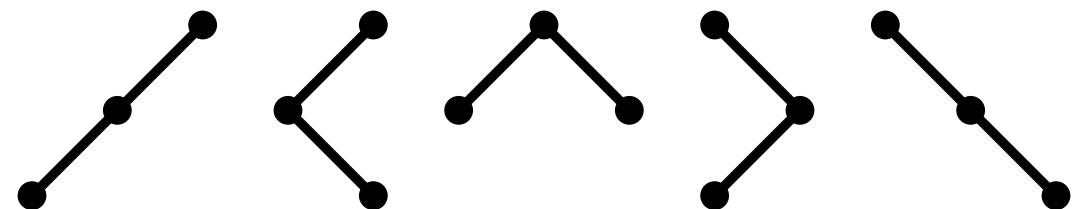
- $b_n \leftarrow$ Número de string binarios
- ¿Cómo obtener relaciones de recurrencia?
- **Usualmente:** particionar por casos y reconocer casos más pequeños.
 - Principio aditivo.
- **Ejemplos:**
 - $e_n \leftarrow$ formas de subir n peldaños con pasos de 1 y 2 peldaños.

Ejemplos

- $b_n \leftarrow$ Número de string binarios
- ¿Cómo obtener relaciones de recurrencia?
- **Usualmente:** particionar por casos y reconocer casos más pequeños.
 - Principio aditivo.
- **Ejemplos:**
 - $e_n \leftarrow$ formas de subir n peldaños con pasos de 1 y 2 peldaños.
 - $t_n \leftarrow$ árboles binarios con n nodos.



$n = 3$



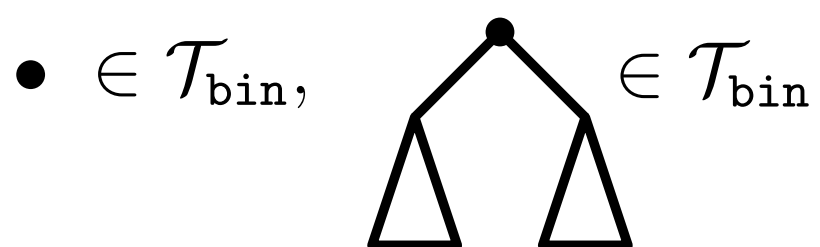
Ejemplos

- $b_n \leftarrow$ Número de string binarios
- ¿Cómo obtener relaciones de recurrencia?
- **Usualmente:** particionar por casos y reconocer casos más pequeños.
 - Principio aditivo.

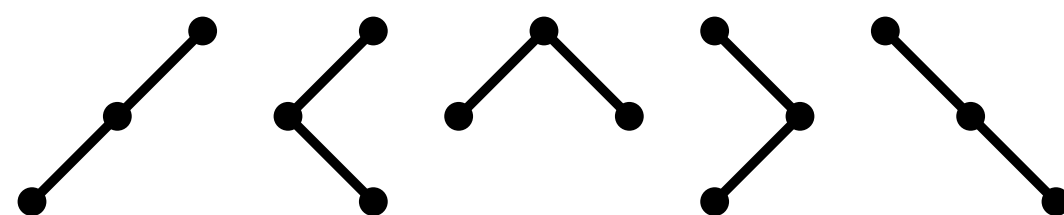
- **Ejemplos:**

- $e_n \leftarrow$ formas de subir n peldaños con pasos de 1 y 2 peldaños.

- $t_n \leftarrow$ árboles binarios con n nodos.



$n = 3$



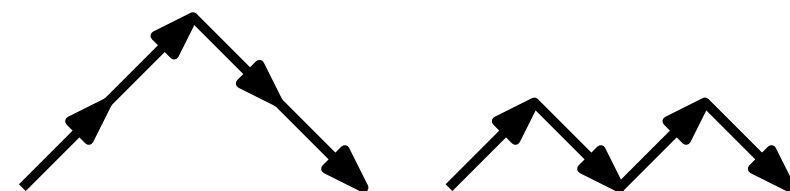
- $d_n \leftarrow$ caminos de Dyck con $2n$ pasos.

usa sólo $\nearrow$ and $\searrow$

sobre la horizontal

parte y termina en la horizontal

$n = 2$



Sistemas de recurrencias

Def.

Un *sistema de recurrencias* es una colección de sucesiones definidas recursivamente de manera *simultánea*.

Sistemas de recurrencias

Def.

Un *sistema de recurrencias* es una colección de sucesiones definidas recursivamente de manera *simultánea*.

Acá, las ecuaciones de una sucesión pueden depender de los términos anteriores de las otras sucesiones del sistema.

Sistemas de recurrencias

Def.

Un *sistema de recurrencias* es una colección de sucesiones definidas recursivamente de manera *simultánea*.

Acá, las ecuaciones de una sucesión pueden depender de los términos anteriores de las otras sucesiones del sistema.

- **Ejemplos:**

$$a_n = b_{n-1} + c_{n-2}$$

$$b_n = \sin c_{\lfloor n/2 \rfloor}$$

$$c_n = a_{n-1} b_{n-1}.$$

Sistemas de recurrencias

Def.

Un *sistema de recurrencias* es una colección de sucesiones definidas recursivamente de manera *simultánea*.

Acá, las ecuaciones de una sucesión pueden depender de los términos anteriores de las otras sucesiones del sistema.

- **Ejemplos:**

$$a_n = b_{n-1} + c_{n-2}$$

$$b_n = \sin c_{\lfloor n/2 \rfloor}$$

$$c_n = a_{n-1} b_{n-1}.$$

- **Útil** cuando una sucesión no guarda toda la información.

Sistemas de recurrencias

Def.

Un *sistema de recurrencias* es una colección de sucesiones definidas recursivamente de manera *simultánea*.

Acá, las ecuaciones de una sucesión pueden depender de los términos anteriores de las otras sucesiones del sistema.

- **Ejemplos:**

$$a_n = b_{n-1} + c_{n-2}$$

$$b_n = \sin c_{\lfloor n/2 \rfloor}$$

$$c_n = a_{n-1} b_{n-1}.$$

- **Útil** cuando una sucesión no guarda toda la información.
- **Ejemplo:** $p_n \leftarrow$ string ternarios de largo n con par 1's.

Fórmulas cerradas

Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?

Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?
 - **Técnicas elementales:** inspección, expansión y cambio de variable.

Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?
 - **Técnicas elementales:** inspección, expansión y cambio de variable.
- **Inspección:**

Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?
 - **Técnicas elementales:** inspección, expansión y cambio de variable.
- **Inspección:**
 - *Adivinar* una solución (Ansatz),

Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?
 - **Técnicas elementales:** inspección, expansión y cambio de variable.
- **Inspección:**
 - *Adivinar* una solución (Ansatz),
 - Probar que es solución via inducción.

Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?
 - **Técnicas elementales:** inspección, expansión y cambio de variable.
- **Inspección:**
 - *Adivinar* una solución (Ansatz),
 - Probar que es solución via inducción.
- **Ejemplo:** torres de Hanoi.



Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?
 - **Técnicas elementales:** inspección, expansión y cambio de variable.
- **Inspección:**
 - *Adivinar* una solución (Ansatz),
 - Probar que es solución via inducción.
- **Ejemplo:** torres de Hanoi.
 - Sólo un disco a la vez



Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?
 - **Técnicas elementales:** inspección, expansión y cambio de variable.
- **Inspección:**
 - *Adivinar* una solución (Ansatz),
 - Probar que es solución via inducción.
- **Ejemplo:** torres de Hanoi.
 - Sólo un disco a la vez
 - No se puede poner un disco sobre uno más pequeño

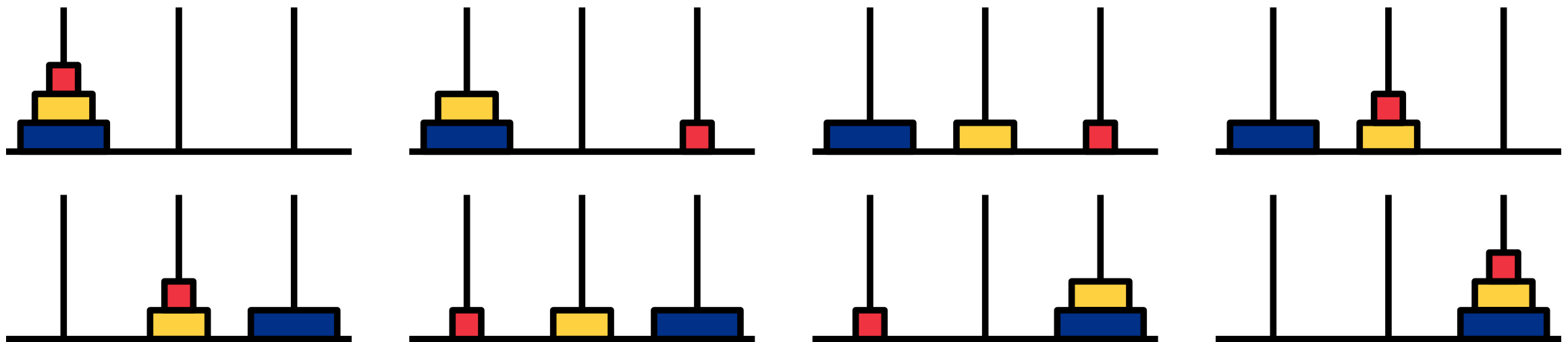


Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?
 - **Técnicas elementales:** inspección, expansión y cambio de variable.
- **Inspección:**
 - *Adivinar* una solución (Ansatz),
 - Probar que es solución via inducción.

- **Ejemplo:** torres de Hanoi.

- Sólo un disco a la vez
- No se puede poner un disco sobre uno más pequeño
- **Obj:** mover todos los discos de A a C .

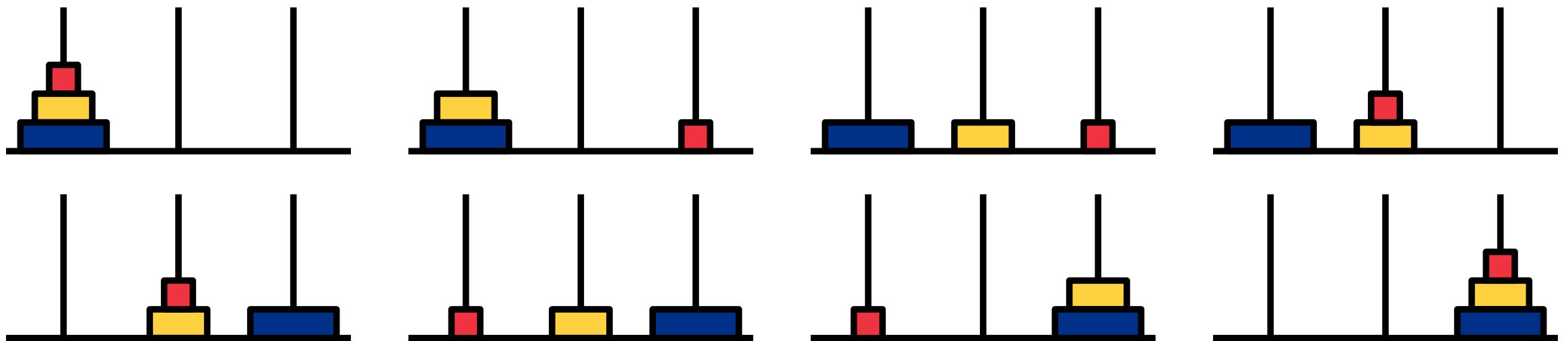


Inspección

- ¿Cómo usar recurrencias para encontrar fórmulas cerradas?
 - **Técnicas elementales:** inspección, expansión y cambio de variable.
- **Inspección:**
 - *Adivinar* una solución (Ansatz),
 - Probar que es solución via inducción.

- **Ejemplo:** torres de Hanoi.

- Sólo un disco a la vez
- No se puede poner un disco sobre uno más pequeño
- **Obj:** mover todos los discos de A a C .



- ¿Con cuántos pasos se puede hacer?

■ Expansión - tipo suma

- **Expansión:**

Expansión - tipo suma

- **Expansión:**
 - *Desenrollar* la recurrencia:

Expansión - tipo suma

- **Expansión:**
 - *Desenrollar* la recurrencia:
 - Aplicar la recurrencia repetidas veces hasta hacer aparecer casos base.

Expansión - tipo suma

- **Expansión:**
 - *Desenrollar* la recurrencia:
 - Aplicar la recurrencia repetidas veces hasta hacer aparecer casos base.
- **Prototipo:** Recurrencia “tipo suma” $a_n = a_{n-1} + f(n)$.

Expansión - tipo suma

- **Expansión:**
 - *Desenrollar* la recurrencia:
 - Aplicar la recurrencia repetidas veces hasta hacer aparecer casos base.
- **Prototipo:** Recurrencia “tipo suma” $a_n = a_{n-1} + f(n)$.

$$a_n = a_{n-1} + f(n)$$

$$= a_{n-2} + f(n-1) + f(n)$$

$$= a_{n-3} + f(n-2) + f(n-1) + f(n)$$

⋮

$$= a_0 + \sum_{k=1}^n f(k).$$

Expansión - tipo suma

- **Expansión:**
 - *Desenrollar* la recurrencia:
 - Aplicar la recurrencia repetidas veces hasta hacer aparecer casos base.

- **Prototipo:** Recurrencia “tipo suma” $a_n = a_{n-1} + f(n)$.

$$\begin{aligned}a_n &= a_{n-1} + f(n) \\&= a_{n-2} + f(n-1) + f(n) \\&= a_{n-3} + f(n-2) + f(n-1) + f(n) \\&\vdots \\&= a_0 + \sum_{k=1}^n f(k).\end{aligned}$$

- Reduce el problema a calcular una suma

Expansión - tipo suma

- **Expansión:**

- *Desenrollar* la recurrencia:
- Aplicar la recurrencia repetidas veces hasta hacer aparecer casos base.

- **Prototipo:** Recurrencia “tipo suma” $a_n = a_{n-1} + f(n)$.

$$\begin{aligned}a_n &= a_{n-1} + f(n) \\&= a_{n-2} + f(n-1) + f(n) \\&= a_{n-3} + f(n-2) + f(n-1) + f(n) \\&\vdots \\&= a_0 + \sum_{k=1}^n f(k).\end{aligned}$$

- Reduce el problema a calcular una suma
- **Ejemplo:**

Expansión - tipo suma

- **Expansión:**

- *Desenrollar* la recurrencia:
- Aplicar la recurrencia repetidas veces hasta hacer aparecer casos base.

- **Prototipo:** Recurrencia “tipo suma” $a_n = a_{n-1} + f(n)$.

$$\begin{aligned}a_n &= a_{n-1} + f(n) \\&= a_{n-2} + f(n-1) + f(n) \\&= a_{n-3} + f(n-2) + f(n-1) + f(n) \\&\vdots \\&= a_0 + \sum_{k=1}^n f(k).\end{aligned}$$

- Reduce el problema a calcular una suma

- **Ejemplo:**

- n Personas llegan una por una a una reunión.
Cada persona saluda a todas las anteriores.
¿Número de saludos?

Expansión - Ejemplo

- Círculo de Moser

Expansión - Ejemplo

- Círculo de Moser
 - n puntos en un círculo

Expansión - Ejemplo

- Círculo de Moser
 - n puntos en un círculo
 - se trazan todas las cuerdas

Expansión - Ejemplo

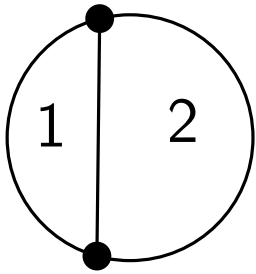
- Círculo de Moser
 - n puntos en un círculo
 - se trazan todas las cuerdas
 - ningún trío de cuerdas se intersectan en el mismo punto

Expansión - Ejemplo

- Círculo de Moser
 - n puntos en un círculo
 - se trazan todas las cuerdas
 - ningún trío de cuerdas se intersectan en el mismo punto
 - ¿Cuántas áreas se forman?

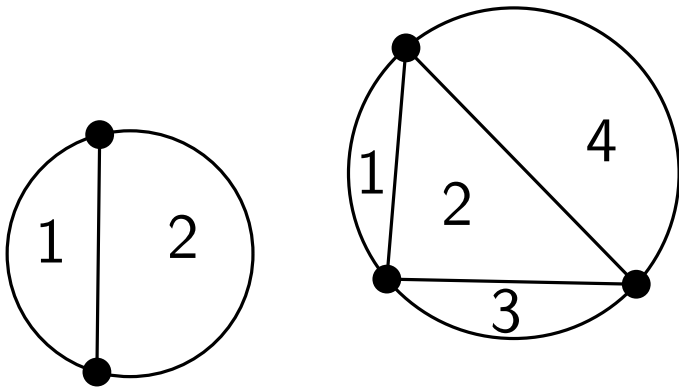
Expansión - Ejemplo

- Círculo de Moser
 - n puntos en un círculo
 - se trazan todas las cuerdas
 - ningún trío de cuerdas se intersectan en el mismo punto
 - ¿Cuántas áreas se forman?



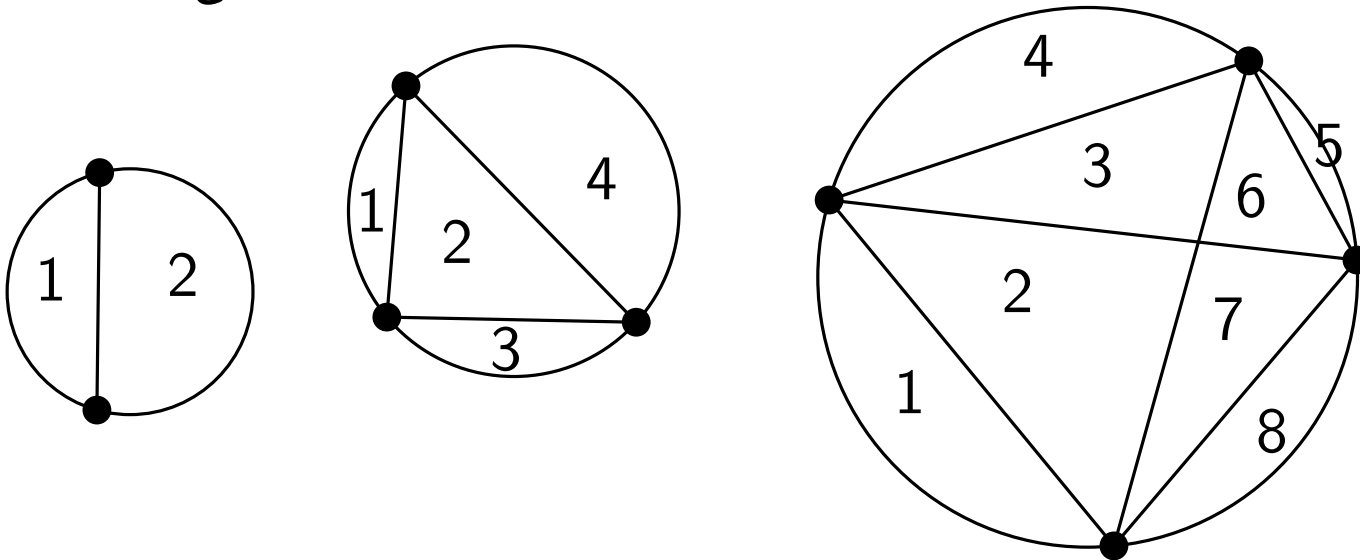
Expansión - Ejemplo

- Círculo de Moser
 - n puntos en un círculo
 - se trazan todas las cuerdas
 - ningún trío de cuerdas se intersectan en el mismo punto
 - ¿Cuántas áreas se forman?



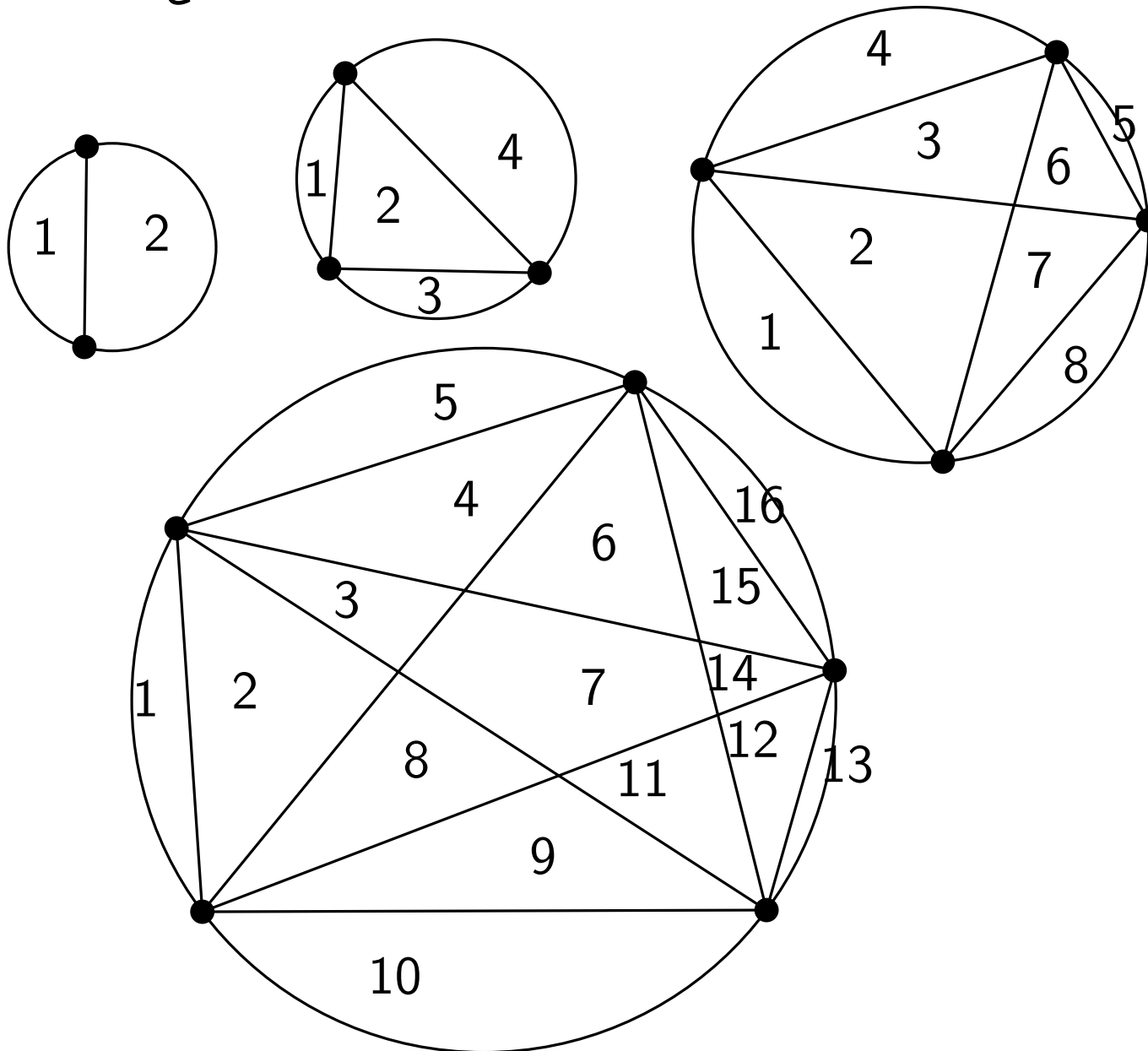
Expansión - Ejemplo

- Círculo de Moser
 - n puntos en un círculo
 - se trazan todas las cuerdas
 - ningún trío de cuerdas se intersectan en el mismo punto
 - ¿Cuántas áreas se forman?



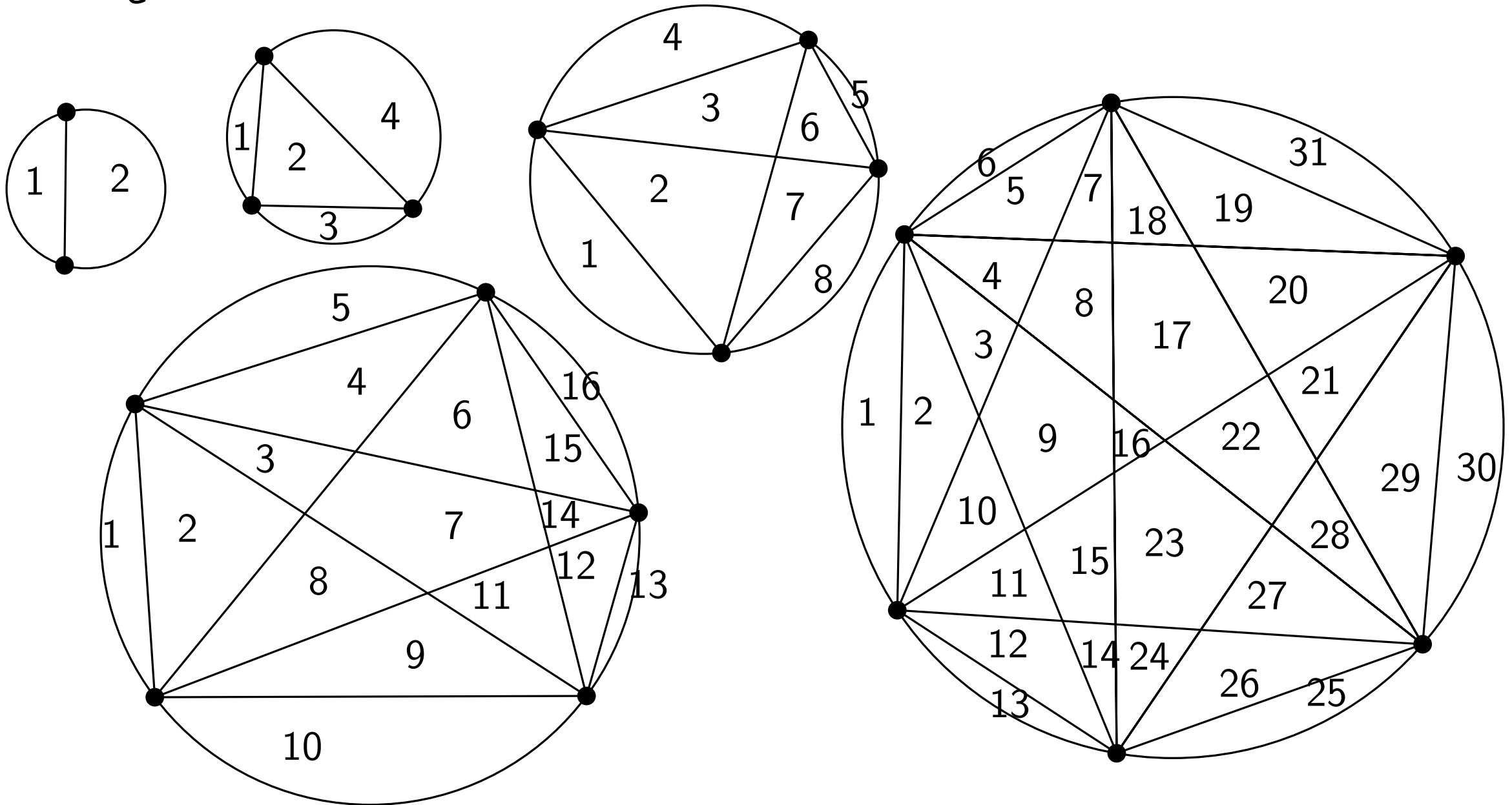
Expansión - Ejemplo

- Círculo de Moser
 - n puntos en un círculo
 - se trazan todas las cuerdas
 - ningún trío de cuerdas se intersectan en el mismo punto
 - ¿Cuántas áreas se forman?



Expansión - Ejemplo

- Círculo de Moser
 - n puntos en un círculo
 - se trazan todas las cuerdas
 - ningún trío de cuerdas se intersectan en el mismo punto
 - ¿Cuántas áreas se forman?



Expansión - Producto

- **Prototipo:** Recurrencia “tipo producto” $a_n = a_{n-1} \cdot f(n)$.

Expansión - Producto

- **Prototipo:** Recurrencia “tipo producto” $a_n = a_{n-1} \cdot f(n)$.

$$a_n = f(n) \cdot a_{n-1}$$

$$= f(n) \cdot f(n-1) \cdot a_{n-2}$$

$$= f(n) \cdot f(n-1) \cdot f(n-2) \cdot a_{n-3}$$

$\vdots$

$$= a_0 \cdot \prod_{k=1}^n f(k).$$

Expansión - Producto

- **Prototipo:** Recurrencia “tipo producto” $a_n = a_{n-1} \cdot f(n)$.

$$a_n = f(n) \cdot a_{n-1}$$

$$= f(n) \cdot f(n-1) \cdot a_{n-2}$$

$$= f(n) \cdot f(n-1) \cdot f(n-2) \cdot a_{n-3}$$

$\vdots$

$$= a_0 \cdot \prod_{k=1}^n f(k).$$

- Reduce el problema a calcular un producto

Expansión - Producto

- **Prototipo:** Recurrencia “tipo producto” $a_n = a_{n-1} \cdot f(n)$.

$$a_n = f(n) \cdot a_{n-1}$$

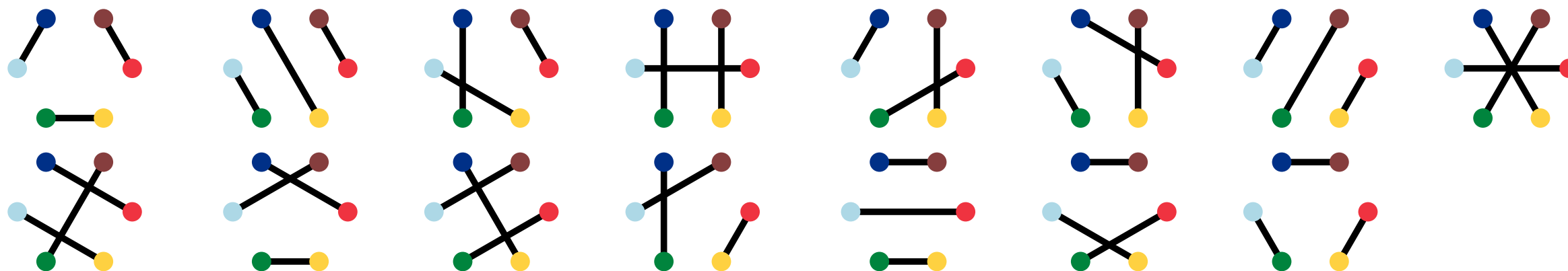
$$= f(n) \cdot f(n-1) \cdot a_{n-2}$$

$$= f(n) \cdot f(n-1) \cdot f(n-2) \cdot a_{n-3}$$

⋮

$$= a_0 \cdot \prod_{k=1}^n f(k).$$

- Reduce el problema a calcular un producto
- **Ejemplo:** emparejar $2n$ personas de manera perfecta.



Expansión - Producto

- **Prototipo:** Recurrencia “tipo producto” $a_n = a_{n-1} \cdot f(n)$.

$$a_n = f(n) \cdot a_{n-1}$$

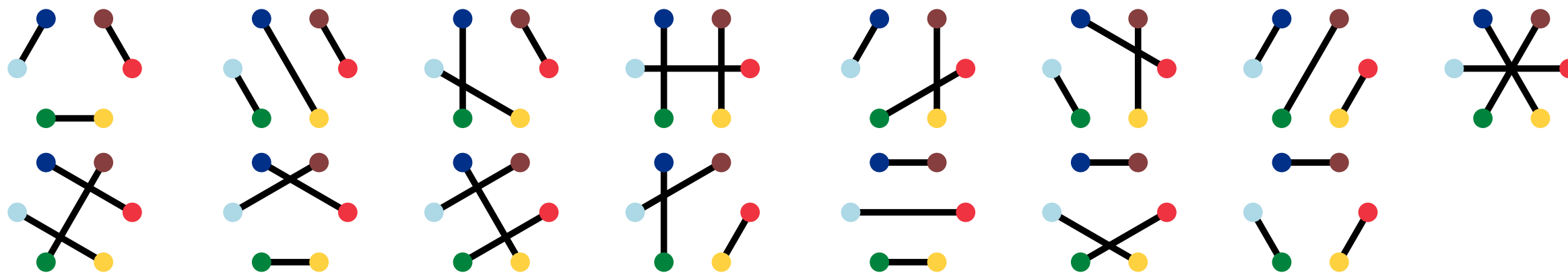
$$= f(n) \cdot f(n-1) \cdot a_{n-2}$$

$$= f(n) \cdot f(n-1) \cdot f(n-2) \cdot a_{n-3}$$

⋮

$$= a_0 \cdot \prod_{k=1}^n f(k).$$

- Reduce el problema a calcular un producto
- **Ejemplo:** emparejar $2n$ personas de manera perfecta.



- **Obs:** hay otras que se resuelven via expansión: $T(n) = 2T(n/2) + C$.

Cambio de variable

- **Cambio de variable:**

Cambio de variable

- **Cambio de variable:**
 - Definir una nueva sucesión c_n a partir de mi sucesión original a_n .

Cambio de variable

- **Cambio de variable:**

- Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
- Encontrar una fórmula explícita para c_n

Cambio de variable

- **Cambio de variable:**

- Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
- Encontrar una fórmula explícita para c_n
- Usar eso para dar una fórmula explícita para a_n .

Cambio de variable

- **Cambio de variable:**
 - Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
 - Encontrar una fórmula explícita para c_n
 - Usar eso para dar una fórmula explícita para a_n .
- **CV clásico:** forzar suma/producto.

Cambio de variable

- **Cambio de variable:**

- Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
- Encontrar una fórmula explícita para c_n
- Usar eso para dar una fórmula explícita para a_n .

- **CV clásico:** forzar suma/producto.

- **Ejemplo:**

Cambio de variable

- **Cambio de variable:**

- Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
- Encontrar una fórmula explícita para c_n
- Usar eso para dar una fórmula explícita para a_n .

- **CV clásico:** forzar suma/producto.

- **Ejemplo:**

- Un virus se cuadriplica cada minuto

Cambio de variable

- **Cambio de variable:**

- Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
- Encontrar una fórmula explícita para c_n
- Usar eso para dar una fórmula explícita para a_n .

- **CV clásico:** forzar suma/producto.

- **Ejemplo:**

- Un virus se cuadriplica cada minuto

Cambio de variable

- **Cambio de variable:**

- Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
- Encontrar una fórmula explícita para c_n
- Usar eso para dar una fórmula explícita para a_n .

- **CV clásico:** forzar suma/producto.

- **Ejemplo:**

- Un virus se cuadriplica cada minuto
- Un equipo lograr reparar 3 unidades infectadas cada minuto

Cambio de variable

- **Cambio de variable:**

- Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
- Encontrar una fórmula explícita para c_n
- Usar eso para dar una fórmula explícita para a_n .

- **CV clásico:** forzar suma/producto.

- **Ejemplo:**

- Un virus se cuadriplica cada minuto
- Un equipo lograr reparar 3 unidades infectadas cada minuto
- Al minuto 0 hay 2 unidades infectadas.

Cambio de variable

- **Cambio de variable:**

- Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
- Encontrar una fórmula explícita para c_n
- Usar eso para dar una fórmula explícita para a_n .

- **CV clásico:** forzar suma/producto.

- **Ejemplo:**

- Un virus se cuadriplica cada minuto
- Un equipo lograr reparar 3 unidades infectadas cada minuto
- Al minuto 0 hay 2 unidades infectadas.

- **Ejemplo:**

Cambio de variable

- **Cambio de variable:**

- Definir una nueva sucesión c_n a partir de mi sucesión original a_n .
- Encontrar una fórmula explícita para c_n
- Usar eso para dar una fórmula explícita para a_n .

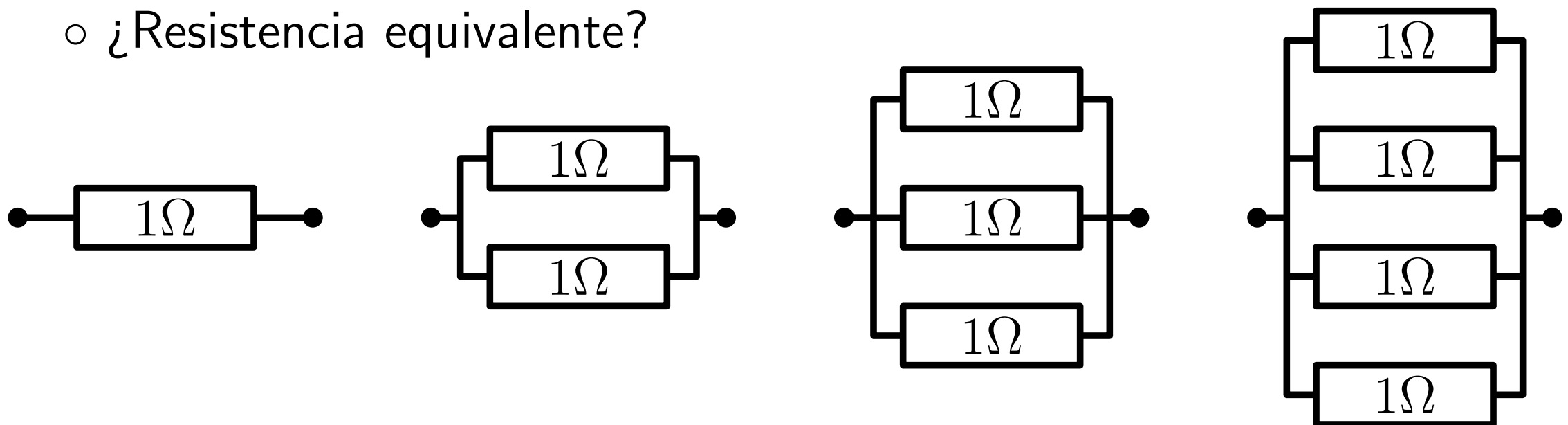
- **CV clásico:** forzar suma/producto.

- **Ejemplo:**

- Un virus se cuadriplica cada minuto
- Un equipo lograr reparar 3 unidades infectadas cada minuto
- Al minuto 0 hay 2 unidades infectadas.

- **Ejemplo:**

- ¿Resistencia equivalente?



Dudas/Preguntas

Wrap-Up

- **Hoy:**

Wrap-Up

- **Hoy:**
 - Recurrencias

Wrap-Up

- **Hoy:**
 - Recurrencias
 - Formas elementales de resolución

Wrap-Up

- **Hoy:**

- Recurrencias
- Formas elementales de resolución
- Inspección, expansión, cambio de variable.

Wrap-Up

- **Hoy:**

- Recurrencias
- Formas elementales de resolución
- Inspección, expansión, cambio de variable.

- **Próx. clase:**

Wrap-Up

- **Hoy:**

- Recurrencias
- Formas elementales de resolución
- Inspección, expansión, cambio de variable.

- **Próx. clase:**

- Funciones generatrices

Wrap-Up

- **Hoy:**

- Recurrencias
- Formas elementales de resolución
- Inspección, expansión, cambio de variable.

- **Próx. clase:**

- Funciones generatrices
- Cómo usarlas para resolver recurrencias