

# Computación Bio–inspirada

## Tema 3: Estructura del ADN. Experimentos de Adleman y de Lipton

David Orellana Martín  
Mario de J. Pérez Jiménez

Grupo de Investigación en Computación Natural  
Dpto. Ciencias de la Computación e Inteligencia Artificial  
Universidad de Sevilla

dorellana@us.es (<http://www.cs.us.es/~dorellana/>)

marper@us.es (<http://www.cs.us.es/~marper/>)

**Máster Universitario en Lógica, Computación e Inteligencia Artificial**  
Curso 2023–2024



# Índice

- \* Estructura del ADN.
- \* Operaciones con moléculas de ADN.
- \* El experimento (**práctico**) de **L. Adleman**.
- \* El experimento (**teórico**) de **R.J. Lipton**.
- \* Máquinas moleculares.

# ¿Modelos de computación **no convencionales**?

¿Porqué son necesarios dichos modelos?

- \* Mejorar la resolución de problemas computacionalmente duros, en el aspecto **cuantitativo**.

¿Qué aportan las moléculas de ADN?

- \* Codifican la **información genética** de los seres vivos.
- \* Tecnología avanzada para **su manipulación**.
- \* **Soporte físico** para realizar computaciones.
- \* Capacidad para implementar un **paralelismo masivo** de forma natural.
- \* Utilizan un alfabeto con cuatro símbolos **A, C, G, T**.

# Molécula de ADN: Un poco de historia

- \* **F. Miescher (1869)** consiguió aislar una molécula de ADN, separando la membrana plasmática de una célula, del citoplasma (por centrifugación).



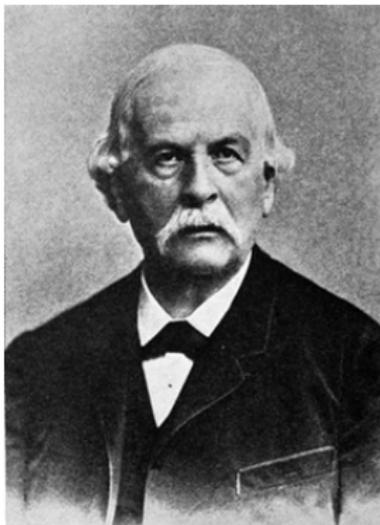
# Molécula de ADN: Un poco de historia

- \* **W. Flemming** (1882) descubrió la estructura y el comportamiento de los **cromosomas** (descubierto por Holfmeister).



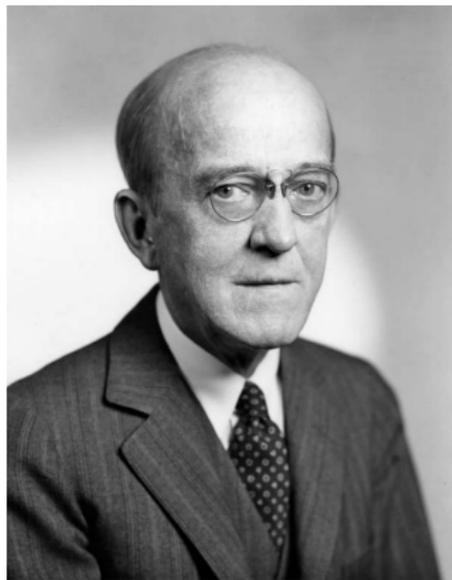
# Molécula de ADN: Un poco de historia

- \* **R. Altmann (1889)** justificó que la molécula de ADN era ácida (denominándola **ácido nucleico**) y sólo se encontraba en los cromosomas.



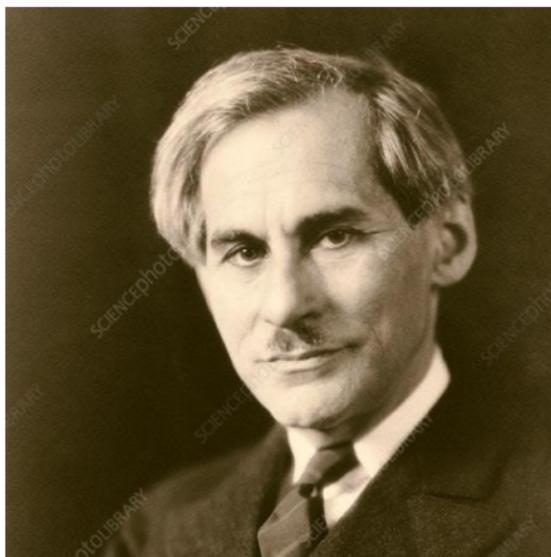
# Molécula de ADN: Un poco de historia

- \* **F. Griffith (1928)** y **O. Avery (1938)** justificaron que las moléculas de ADN eran la base de la herencia genética.



# Molécula de ADN: Un poco de historia

- \* **Ph. Levene (1929)** identificó las componentes de las moléculas de ADN (fosfato, azúcar desoxirribosa y cuatro bases nitrogenadas) unidas en ese "orden": fosfato-azúcar-base (**nucleótido**).

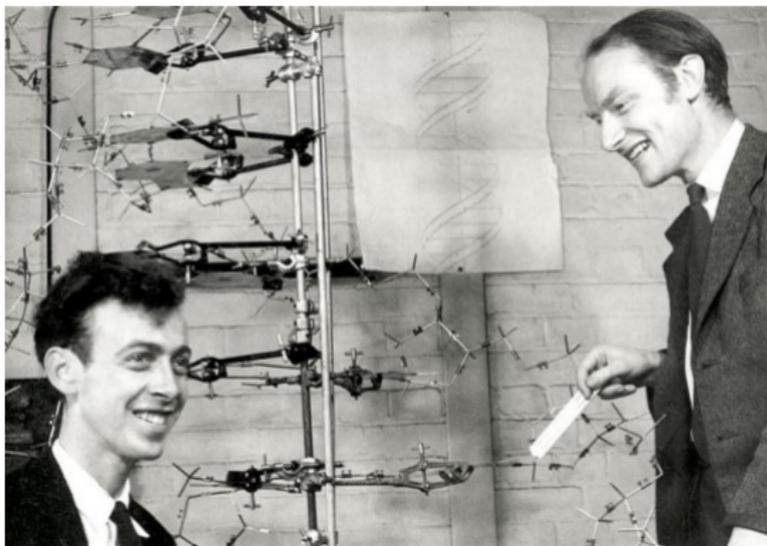


# Molécula de ADN: Un poco de historia

- \* **R. Franklin** (1952) cristalizó una molécula de ADN, consiguiendo la foto más clara, hasta entonces, acerca de su estructura.



- \* **1953:** **J. Watson** y **F. Crick** desarrollaron un modelo tridimensional de la estructura del ADN (sintetizando trabajos previos) y enunciaron el principio de **complementariedad** (Premio Nobel de Medicina en 1962).

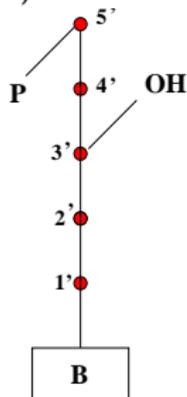


# La molécula de ADN



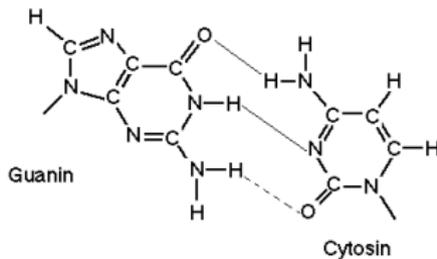
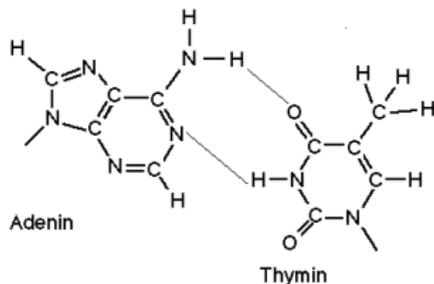
# Estructura del ADN

- Polímero que consta de una serie de monómeros (**nucleótidos**).
- Cada nucleótido consta de:
  - ★ Un azúcar con cinco átomos de carbono (*desoxirribosa*).
  - ★ Un grupo fosfato (**P**), situado en el carbono 5'.
  - ★ Un grupo hidroxilo (**OH**), situado en el carbono 3'.
  - ★ Una base nitrogenada (**B**).



# Estructura del ADN

Bases nitrogenadas: **A** (adenina), **C** (citosina), **G** (guanina) y **T** (timina).

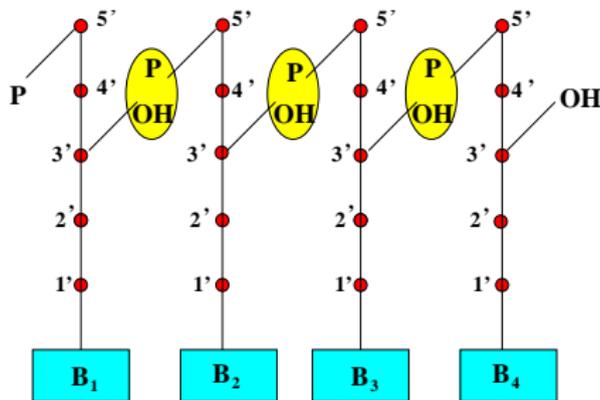


★ Purinas: **A** y **G**.

★ Pirimidinas: **C** y **T**.

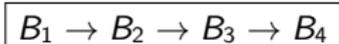
# Cadenas (hebras) simples de ADN

**Enlace fosfodiéster:** *cadenas/hebras simples.*



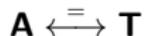
**Orientación** de las cadenas simples de ADN:

★ La **dirección de la Vida**: 5' – 3'.



# Cadenas (hebras) dobles de ADN

Enlace de hidrógeno:

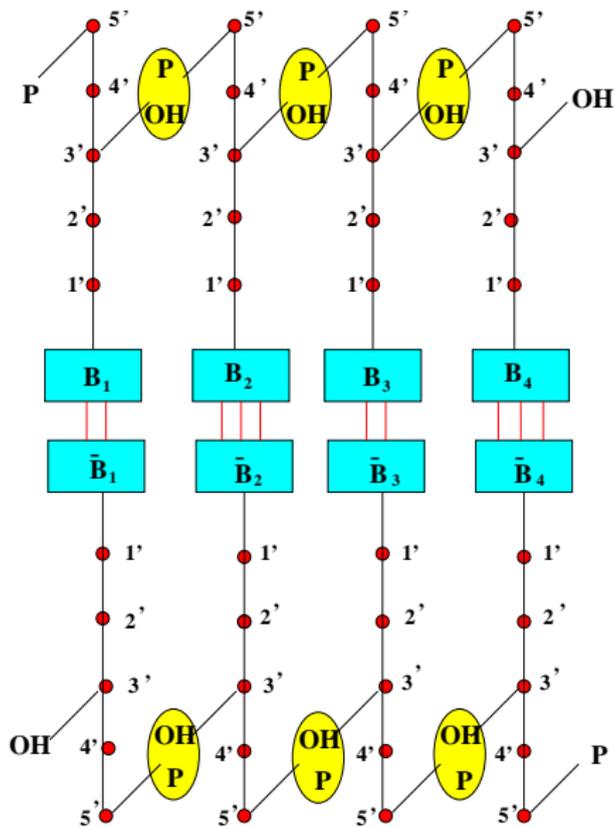


complementariedad de Watson-Crick

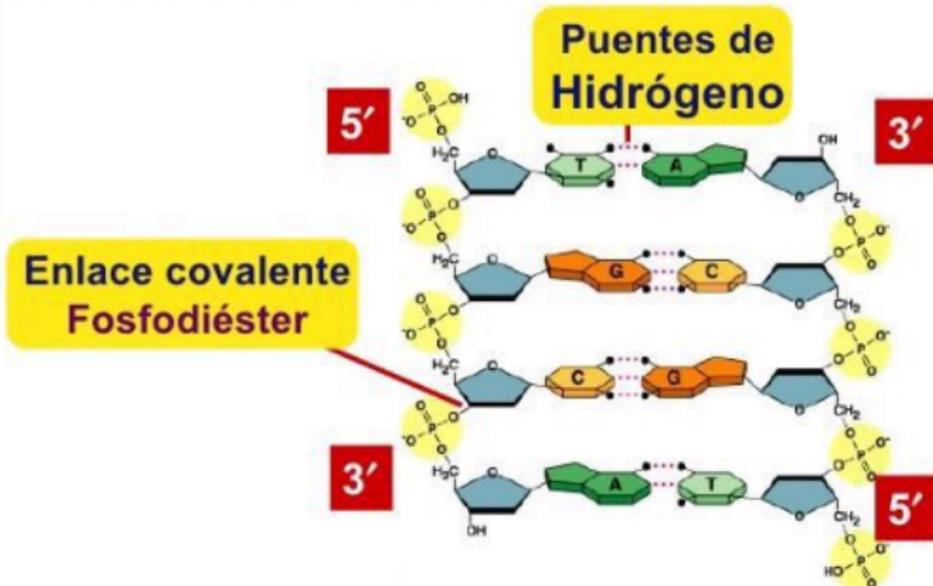
Cadenas dobles, doble hebras (estructura de *doble hélice*):

- Enlaces fosfodiéster + enlaces de hidrógeno.
- Dos cadenas simples con orientaciones opuestas.

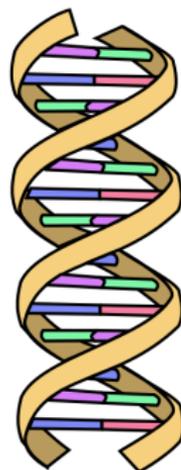
# Cadenas (hebras) dobles de ADN



# Enlaces en el DNA



# Estructura de doble hélice



■ = Adenina

■ = Timina

■ = Citosina

■ = Guanina

■ = Esqueleto  
azucar-fosfato

ADN



Estructura helicoidal de una molécula de ADN

# Una “cremallera” muy especial



Ruptura de los puentes de hidrógeno

# Hacia un modelo de computación molecular

**Estructura de datos:** moléculas de ADN.

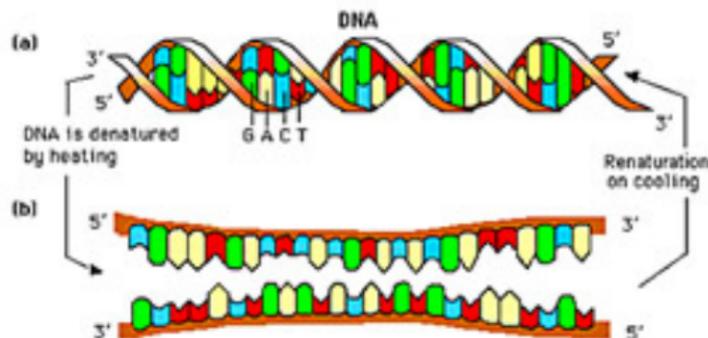
**Cuatro operaciones** básicas con moléculas de ADN:

- **Desnaturalización:**

- \* Entrada (T): hebras dobles.
- \* Calentamiento lento:  $\sim 85-94$  grados C.
- \* Salida (T'): hebras simples.

- **Renaturalización.**

- \* Entrada (T): hebras simples.
- \* Enfriamiento lento:  $\sim 45-55$  grados C.
- \* Salida (T'): hebras dobles.



# Hacia un modelo de computación molecular

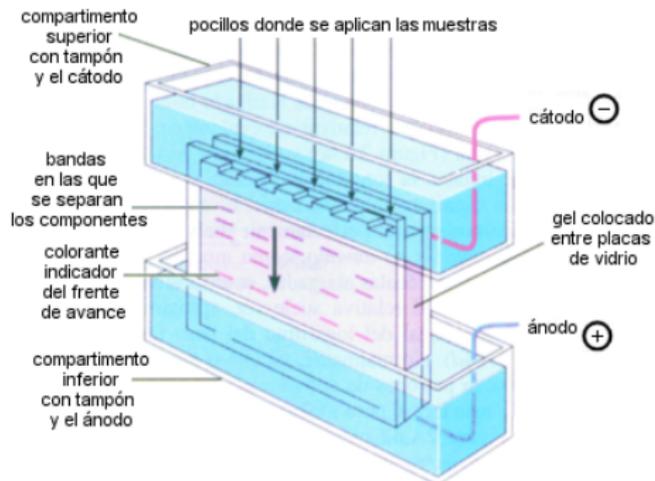
- **Medida de la longitud** (Unidades de medida: *mer* y *bp*).

- \* Entrada: Un tubo de ensayo *T* con doble hebras de ADN.

- \* Salida: las longitudes de las moléculas de ADN.

- \* Técnica de la **Electroforesis en gel (agarosa)**.

- ★ Bases nitrogenadas: carga negativa proporcional a su longitud.



# Hacia un modelo de computación molecular

- **Extracción.**

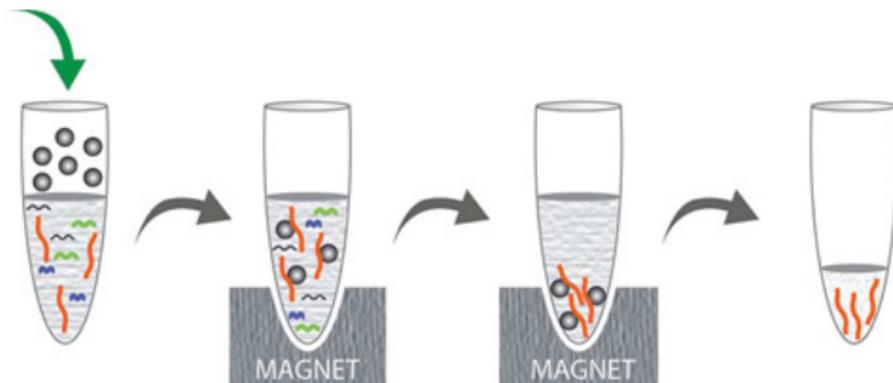
- \* Entrada:  $(T, \gamma)$ ,  $T$  tubo de ensayo con moléculas de ADN y  $\gamma$  un "oligonucleótido" (hebra corta de ADN).
- \* Salida: dos tubos de ensayo  $+(T, \gamma)$  y  $-(T, \gamma)$ .
- \* Método del **filtro**.



# Hacia un modelo de computación molecular

- **Extracción.**

\* Método de las **sondas metálicas**.



# Hacia un modelo de computación molecular

**Otras operaciones** relevantes con moléculas de ADN:

- **Copiar y alargar** (*Enzima Polimerasa*).
- **Amplificar** (copias masivas: **PCR**, **Kary Mullis**, 1985).
- **Sintetizar**.
- **Alterar, Cortar** (*Enzima Nucleasa*: enzima de restricción).
- **Empastar** (*Enzima Ligasa*).
- **Leer**.

# Kary Mullis (1944 – 2019)



Licenciado en Química por el *Georgia Institute of Technology*  
Doctor en Química por la *Universidad de California, Berkeley*

# La técnica del PCR: Polymerase Chain Reaction

En 1983, Mullis trabajaba en la compañía biotecnológica **Cetus Corporation** como sintetizador de oligonucleótidos.

- \* Un viernes de abril por la noche, hacía el final de la primavera, mientras conducía por las montañas del norte de California ...
- \* Fracásó en los primeros experimentos realizados (septiembre y octubre de 1983).
- \* Primer experimento con éxito: realizado el 16 de diciembre de 1983 (fragmentos de 25 bp).
- \* Difícil verano de 1984 en **Cetus Corporation**:
  - No se consideraba prioritario desviar recursos al desarrollo de la PCR, aunque ...
  - Se relevó a Mullis como jefe de laboratorio a fin de que explorara a fondo su idea.
  - Se constituyó el “**grupo de la PCR**”, incorporando a cinco miembros de la compañía.

# La técnica del PCR

- \* Estrategia general de la compañía: **patentar** y, después, **publicar** resultados.
- \* 28 de marzo de 1985: primera solicitud de patente (terrible batalla judicial).
- \* Octubre de 1985: Presentación de resultados de la PCR en la *American Society for Human Genetics*.
- \* 20 de diciembre de 1985: publicación en la revista *Science* de las aplicaciones prácticas.
- \* La parte teórica fue aceptada en mayo de 1985 por *Methods of Enzymology* y publicada en 1987 (pero había sido rechazada por *Nature* en diciembre de 1985, y por *Science* en enero 1986).

# Una máquina PCR ...

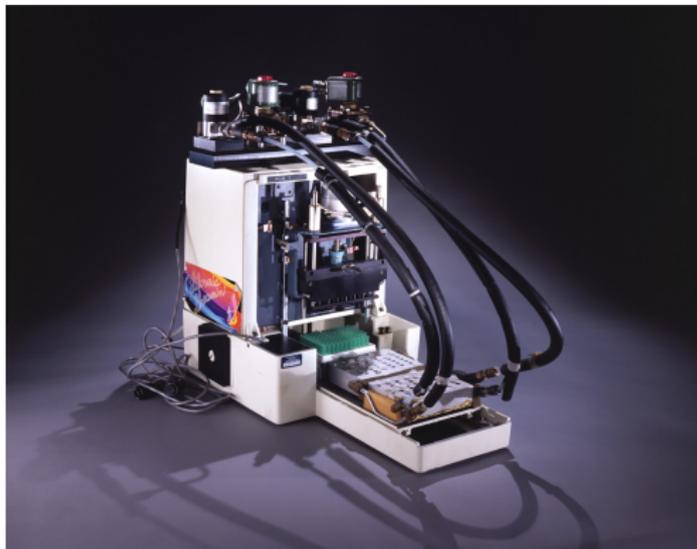
Especie de fotocopiadora de moléculas de ADN:

- \* Las doble hebras de ADN se estructuran en una doble hélice, a modo de una **cremallera** singular.
- \* Aplicando **calor** (lentamente) se puede “**abrir**” la cremallera y usar cada hebra como molde para una copia.
- \* A partir de cada hebra, aplicando **frío** (lentamente) podremos obtener una nueva doble hebra (disponiendo de suficientes nucleótidos).
- \* Y así sucesivamente, mediante ciclos que constan de calentamiento (lento) seguido de enfriamiento (lento).
- \* A lo largo del proceso hay que disponer de nucleótidos en cantidad y de una máquina molecular que los “engarce” (**ADN-polimerasa**).

Obviamente, esa **ADN-polimerasa** debía ser resistente a altas temperaturas.

# Primera máquina que implementa la PCR

**Mr. Cycle:** construida por K. Mullis y su equipo en la **Cetus Corporation** (1985)



Usaba como “máquina molecular” las ADN polimerasas de la bacteria **Escherichia coli** (bacteria del tracto intestinal de animales: sólo soporta temperaturas moderadas).

# Máquinas que implementan la PCR

- \* Para que las máquinas PCR fuesen “**eficientes**”, necesitaban ADN polimerasas resistentes a altas temperaturas (hasta 95<sup>o</sup> C).
- \* En 1965, T. Crook había descrito la enzima *Taq polimerasa* de la bacteria *Thermus Aquaticus* (vivía en las profundidades de manantiales de agua caliente).
- \* En otoño de 1985: S. Stoffel y D. Gelfand del grupo PCR de **Cetus Corporation**, aislaron dicha enzima (en 1989, *Science* la nomina *Molécula del Año*).

Automatización completa de las máquinas PCR.

# Máquinas que implementan la PCR



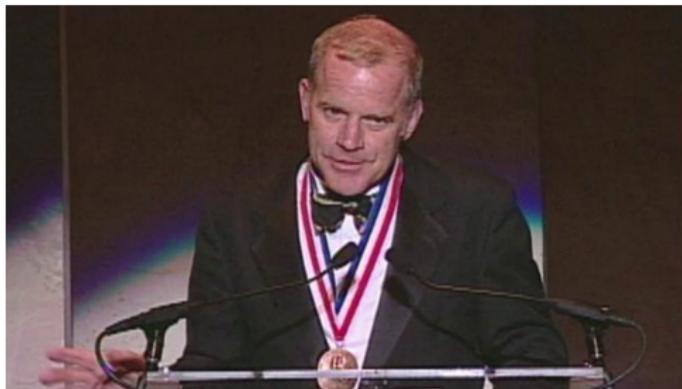
# Y, lo que son las cosas de la Vida ...

Miembros relevantes de la comunidad científica: la técnica del **PCR** era tan sencilla y natural que no podía considerarse como un descubrimiento ni patentarse.

- \* Primavera de 1986: **Cetus Corporation** abonó **10.000 dólares** a K. Mullis por haber desarrollado la técnica del **PCR**.
- \* Marzo de 1989: **Cetus Corporation** obtuvo la patente (¡cuatro años después!)
- \* El 11 de diciembre de 1991, **Cetus Corporation** vendió la patente a la compañía *Hoffmann-La Roche* por **330 millones de dólares**.

K. Mullis perdió dinero, muchísimo dinero, pero ...

# ¡Consiguió la gloria ...!



1993: Premio Nobel de Química (compartido con Michael Smith)

En el discurso posterior la entrega del premio:

- \* No habló de la técnica del **PCR**.
- \* Trató de explicar por qué el virus VIH no era el responsable del SIDA (y que, según él, no era una enfermedad infecciosa) poniendo, además, en cuestión la teoría acerca del cambio climático.

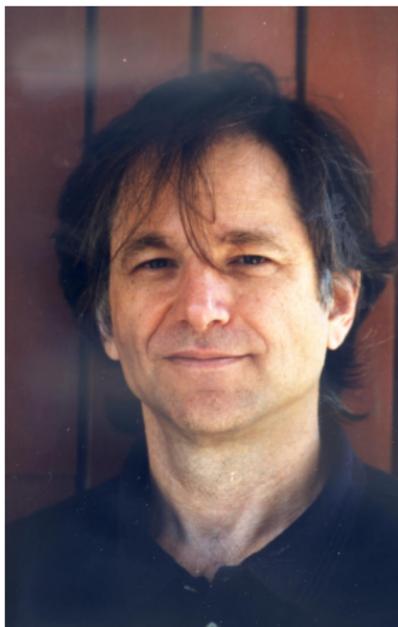
# Técnicas de detección del SARS-CoV-2

**Prueba del PCR:** A partir de una muestra (naso faríngea), permite detectar la existencia de **trozos de material genético** procedente de un cierto virus o de otro agente patógeno.

**Test de diagnóstico rápido:** A partir de una muestra, permite detectar si el organismo

- \* ha desarrollado unos **anticuerpos**, como respuesta del sistema inmune al ataque de un agente invasor (**prueba serológica**, análisis de sangre);
- \* o bien si contiene ciertas **proteínas** producidas por un agente invasor (**prueba de antígenos**, zona posterior de la faringe).

# Leonard M. Adleman (1945 ...)



Licenciado en Matemáticas por la *Universidad de California, Berkeley*, 1968

Doctor en Computer Science por la *Universidad Sur de California*, 1976

Premio Turing, 2002

# El sistema de encriptación RSA: 1978

- \* Tesis doctoral (1976): *Number theoretic aspects of computational complexities* (dirigida por M. Blum).
- \* Noviembre de 1976: Fundamentos teóricos de los sistemas de encriptación con clave pública (W. Diffie y M. Hellman).
- \* 1977: Profesor auxiliar de Matemáticas en el MIT.
- \* R. Rivest y A. Shamir (que trabajaban en el MIT) trataban de encontrar una función para implementar la encriptación con clave pública.
- \* Le propusieron trabajar con ellos, pero la idea no le entusiasmó: aún así, decidió colaborar como “*decodificador*”.

# El sistema de encriptación RSA: 1977-1978

- \* Hasta que en el sistema 43 propuesto por Rivest y Shamir ...
- \* Nace el **sistema RSA** ¿por qué no se llamo sistema **ARS**?
- \* Publicación: *Technical Report* del MIT en 1977 y artículo de *Communications of the ACM*, en 1978<sup>1</sup>.
- \* 1983: se patenta su código y constituyen una empresa (*RSA Data Security Inc.*)
- \* 1996: venden la empresa por 200 millones de dólares.

---

<sup>1</sup>R.L. Rivest, A. Shamir, L.M. Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, **Communications of the ACM**, Vol. 21 (1978), pp 120-126.

# Virus informático

- \* 1980: Regresa a la Universidad Sur de California (SCU) como profesor.
- \* 1983: **Fred Cohen**, alumno de doctorado de Adleman, le propuso la idea de diseñar un programa que “infectara” a otros programas.
- \* Adleman se dio cuenta de la importancia de la idea y propuso el nombre de **computer virus (virus informático)**: el primero se diseñó en noviembre de 1983 y se publicó en 1984 <sup>2</sup>.
- \* Interés por la inmunología: *inusual belleza a los ojos de un matemático*.
- \* Comienza a estudiar los *linfocitos (células T: son de dos tipos CD4 y CD8)*.
- \* Teoría acerca de la enfermedad del SIDA: no aceptada por la comunidad científica.

---

<sup>2</sup>F. Cohen. Computer Viruses. Theory and Experiments, **Computers and Security**, 6 (1987), pp 22-35, North-Holland.

# Nuevos intentos

- \* Para tratar de demostrar su teoría, empezó a “curiosear” en el laboratorio de biología molecular de la SCU (N. Chelyapov).
- \* La ADN polimerasa como una cabeza lectora/escritora de una máquina de Turing.
- \* Objetivo: construir una máquina molecular basada en ADN que simulara una máquina de Turing.
- \* Nuevo fracaso (en 2001, el equipo de E. Shapiro construyó una tal máquina).

Cuando, a principio de los setenta, comenzó sus estudios en la Universidad del Estado de San Francisco afirmaba:

“No me gusta nada hacer experimentos, sólo pensar y razonar sobre las cosas”

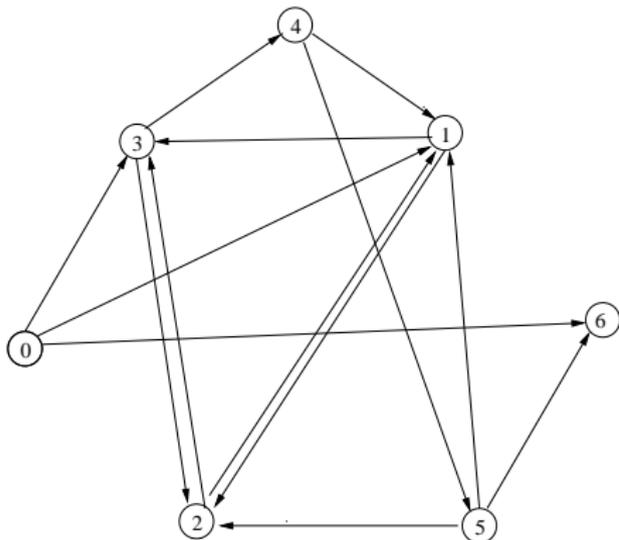
Y las vueltas que da la Vida ...



# El experimento de L. Adleman

Noviembre de 1994:

- \* Resolución molecular de una instancia del **problema del camino hamiltoniano**, en su versión dirigida y con dos nodos distinguidos.



Grafo usado en el experimento de Adleman

El experimento de L. Adleman:

- \* Primer ejemplo de computación a nivel molecular.
- \* Nuevas perspectivas de las moléculas de ADN como **estructura de datos** peculiares.
- \* Posibilidad de usar el **ADN** para resolver instancias de problemas computacionalmente intratables.
- \* Capacidad del ADN para simular computaciones de forma **masivamente paralelas**.

# Implementación en el laboratorio del algoritmo

Mediante la manipulación de moléculas de ADN, se trata de implementar el siguiente algoritmo de *fuerza bruta*:

**Entrada:**  $G = (V, E)$  grafo dirigido;  $v_i$  y  $v_f \in V$

Generar todos los caminos de  $G$ .

Rechazar aquellos que no empiezan por  $v_i$  o no terminan en  $v_f$ .

Rechazar aquellos que no contienen exactamente  $|V|$  nodos.

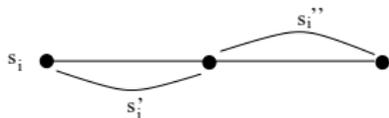
Para cada  $u \in V \setminus \{v_i, v_f\}$ , rechazar aquellos que no contienen  $u$ .

**Salida:** **SÍ**, si queda algún camino; **NO**, en caso contrario.

En el experimento de Adleman,  $v_i = 0$  y  $v_f = 6$ .

# Preparación del tubo de ensayo inicial

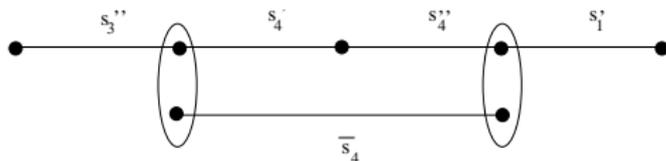
- \* A cada  $i$  ( $0 \leq i \leq 6$ ) se le asocia un oligo  $s_i$  de longitud 20 mer.
- \* Notaremos  $s_i = s'_i s''_i$ , en donde  $|s'_i| = |s''_i| = 10$ .



- \* A cada arco  $(i, j) \in E$  se le asocia el oligo

$$e_{ij} = \begin{cases} s''_i s'_j & \text{si } i \neq 0 \wedge j \neq 6 \\ s_0 s'_j & \text{si } i = 0 \wedge j \neq 6 \\ s''_i s_6 & \text{si } i \neq 0 \wedge j = 6 \\ s_0 s_6 & \text{si } i = 0 \wedge j = 6 \end{cases}$$

- \* Se codifican caminos mediante doble hebras de **ADN** (usando oligos  $\overline{s}_i$ ).  
Ejemplo: *Camino*  $3 \rightarrow 4 \rightarrow 1$



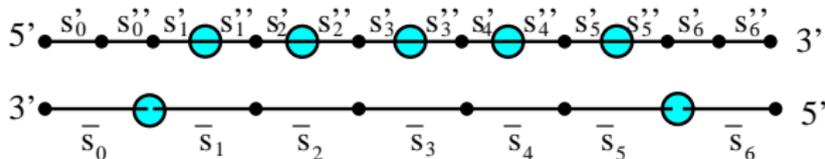
# Implementación de los pasos del algoritmo

## 1.- Generar todos los caminos de $G$ .

- \* Se añade a la solución una cantidad de oligos  $\overline{s}_i$  (50 pmoles) y la misma de oligos  $e_{ij}$ .
- \* La solución se somete a un proceso de renaturalización y empaste.

## 2.- Rechazar los caminos que no empiezan por $v_i$ o no terminan en $v_f$ .

- \* Se aplica la técnica del **PCR** usando como cebadores  $s_0$  y  $\overline{s}_6$ .
- \* Así, sólo serán amplificados los caminos que comienzan por el vértice 0 y terminan por el vértice 6.
- \* El “rechazo” es indirecto: existen muchísimos más caminos que comienzan por el vértice 0 y terminan por el vértice 6.



# Implementación de los pasos del algoritmo

## 3.- Rechazar los caminos que no contienen, exactamente, $|V|$ nodos.

- \* Se implementa utilizando la técnica de la electroforesis en gel: se seleccionan las moléculas de longitud 140 bp.

## 4.- Para cada $u \in V \setminus \{v_i, v_f\}$ , rechazar los caminos que no contienen el vértice $u$ .

- \* Para cada  $i$  ( $1 \leq i \leq 5$ ) se extraen las moléculas que contienen al oligo  $s_i$  (obsérvese que, por el paso 2, no es necesario considerar aquí los casos  $i = 0$  e  $i = 6$ ).

## 5.- Devolver **SÍ**, si queda algún camino, y **NO** en caso contrario.

- \* Para detectar si hay algún camino en el tubo de ensayo de salida, se amplifica el producto obtenido mediante la técnica **PCR** y se hacen circular las moléculas obtenidas a través de un gel.

## Consideraciones acerca del experimento de Adleman (I)

- ★ Procedimiento basado en filtrados.
- ★ Ejecución simultánea de operaciones moleculares.
- ★ Tubo inicial: **número exponencial de cadenas.**
- ★ **Número lineal de operaciones moleculares.**
- ★ Aparecen errores que pueden ser controlados.
- ★ Boneh, Dunworth y Lipton (1995): hasta  $10^{21}$  moléculas de ADN se pueden procesar.
- ★ Realización del experimento de Adleman: hasta grafos con a lo sumo 70 nodos.
- ★ Experimento de Adleman: ¿grafos con 200 nodos y densidad media?

## Consideraciones acerca del experimento de Adleman (II)

- \* Ventajas potenciales:
  - ★ **Velocidad de cálculo:**  $1'2 \times 10^{18}$  versus  $10^{12}$ .
  - ★ **Densidad de información:** 1 bit por  $\text{nm}^3$  versus 1 bit por  $10^{12} \text{ nm}^3$ .
  - ★ **Consumo de energía:** Con un julio de trabajo,  $2 \times 10^{19}$  versus  $10^9$  (del segundo principio de la termodinámica resulta que el máximo número de operaciones irreversibles es  $34 \times 10^{19}$ ).
- \* Nacimiento de la computación ADN.
- \* **No** proporciona un esquema algorítmico.

# El problema SAT

**Definición:** El **lenguaje de la Lógica Proposicional** consta de:

- ★ Conjunto numerable,  $VP$ , de variables proposicionales.
- ★ Conectivas lógicas: " $\neg$ " (negación) y " $\vee$ " (disyunción).
- ★ Símbolos auxiliares: "(" y ")".

**Definición:** El conjunto  $PForm$  de las **fórmulas proposicionales** es el menor conjunto,  $\Gamma$ , que contiene a  $VP$  y

- ★ Si  $P \in \Gamma$ , entonces  $\neg P \in \Gamma$ .
- ★ Si  $P, Q \in \Gamma$ , entonces  $(P \vee Q) \in \Gamma$ .

A partir de  $\neg$  y  $\vee$  se definen las conectivas lógicas  $\wedge$ ,  $\rightarrow$  y  $\leftrightarrow$ .

Notaremos

- ★  $\neg P$  así  $\bar{P}$ .
- ★  $P \vee Q$  así  $P + Q$ .
- ★  $P \wedge Q$  así  $P \cdot Q$ .



**Definición:** Un **literal** es una variable proposicional o la negación de una variable proposicional.

**Definición:** Una **cláusula** es la disyunción de un número finito de literales.

**Definición:** Una fórmula proposicional está en **forma normal conjuntiva (FNC)** si es la conjunción de un número finito de cláusulas.

Toda fórmula proposicional en FNC es la conjunción de una disyunción de literales (las supondremos en **forma "simplificada"**: las cláusulas y las variables no aparecerán repetidas ni serán complementarias).

**Definición:** Una **valoración (asignación) de verdad** es una aplicación de  $VP$  en  $\{0, 1\}$ .

- ★ Toda valoración de verdad se extiende de manera natural a una aplicación de  $PForm$  en  $\{0, 1\}$  (a través de las **tablas de verdad**).

Valoraciones de verdad **relevantes** para una fórmula.

- ★ Una valoración relevante para una fórmula es una aplicación del conjunto de sus variables en  $\{0, 1\}$ .
- ★ Si  $\varphi$  tiene  $n$  variables, entonces el número total de valoraciones relevantes para  $\varphi$  es  $2^n$ .

**Definición:** Dos fórmulas proposicionales son **semánticamente equivalentes** **sii** cualquier valoración le asigna a ambas el mismo valor.

- \* Toda fórmula proposicional tiene una fórmula semánticamente equivalente en FNC (y en forma simplificada).

**Definición:** Una fórmula proposicional,  $\varphi$ , es **satisfactible** **sii** existe, al menos, una valoración,  $\sigma$ , tal que  $\sigma(\varphi) = 1$ .

**Problema SAT:** Dada una fórmula proposicional en FNC (y en forma simplificada), **determinar si es satisfactible**.

- \* Se prueba fácilmente que el problema **SAT** pertenece a la clase **NP**.

**Teorema:** (S.A. Cook, 1971) *El problema **SAT** es **NP**-duro* (más difícil que cualquier problema de **NP**).

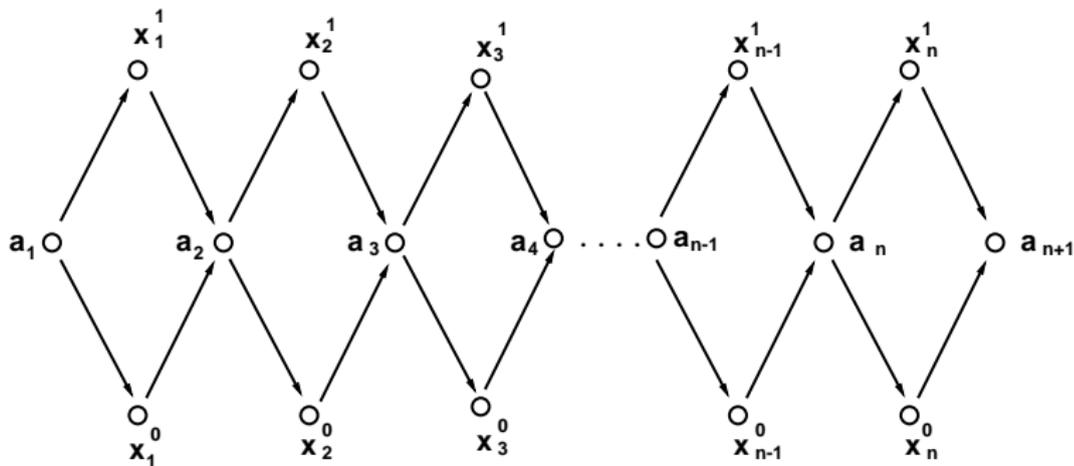
**Consecuencia:** El problema **SAT** es **NP**-completo.

# Experimento de Lipton

**Abril 1995:** R.J. Lipton resuelve una instancia genérica del *problema SAT*.

Sea  $\varphi \equiv c_1 \wedge \dots \wedge c_p$  con  $c_i = l_{i,1} \vee \dots \vee l_{i,r_i}$ , y conjunto de variables

$\text{Var}(\varphi) = \{x_1, \dots, x_n\}$ . Se le asocia un grafo dirigido  $G_n = (V_n, E_n)$ :



Grafo dirigido asociado a una fórmula proposicional con  $n$  variables

# Experimento de Lipton

El grafo  $G_n$  verifica las siguientes propiedades:

- Existen  $2^n$  caminos desde  $a_1$  hasta  $a_{n+1}$ .
- Existe una biyección entre el conjunto de caminos y las valoraciones relevantes para  $\varphi$ .
- ★ El camino  $(a_1, x_1^{j_1}, a_2, x_2^{j_2}, \dots, a_n, x_n^{j_n}, a_{n+1})$  tiene asociado biyectivamente la valoración  $\sigma$  caracterizada por:

$$\left. \begin{array}{l} \sigma(x_1) = j_1 \\ \sigma(x_2) = j_2 \\ \vdots \\ \sigma(x_n) = j_n \end{array} \right\}$$

# Experimento de Lipton

Usa las ideas de Adleman.

Procedimiento basado en filtrados.

Primer esquema algorítmico molecular.

# Notaciones en el experimento de Lipton

Sea  $\varphi$  una fórmula proposicional en forma normal conjuntiva tal que:

- ★  $\varphi \equiv c_1 \wedge \dots \wedge c_p$  ( $c_i$  cláusulas);  $c_i = l_{i,1} \vee \dots \vee l_{i,r_i}$  ( $l_{i,j}$  literales).
- ★  $\text{Var}(\varphi) = \{x_1, \dots, x_n\}$  (variables).

Para cada literal,  $l_{i,j}$ , que aparece en la fórmula  $\varphi$ :

- \* Si  $l_{i,j} = x_m$ , entonces notaremos  $l_{i,j}^1 = x_m^1$ ,  $l_{i,j}^0 = x_m^0$ .
- \* Si  $l_{i,j} = \bar{x}_m$ , entonces notaremos  $l_{i,j}^1 = x_m^0$ ,  $l_{i,j}^0 = x_m^1$ .

Además:

- \* Las valoraciones  $\sigma$  relevantes para  $\varphi$  se representan mediante moléculas de ADN (**caminos** en el grafo desde  $a_1$  hasta  $a_{n+1}$ ).
- \*  $l_{i,j}^1$  y  $l_{i,j}^0$  se representan mediante moléculas de ADN.
- \* Si  $\sigma$  contiene a  $l_{i,j}^1$  como subcadena, entonces  $\sigma(l_{i,j}) = 1$ .
- \* Si  $\sigma$  contiene a  $l_{i,j}^0$  como subcadena, entonces  $\sigma(l_{i,j}) = 0$ .

# Idea del experimento de Lipton

A partir de una fórmula de entrada, se procede como sigue:

- Se parte del **tubo inicial**  $T_0$  que codifica todas las valoraciones relevantes para  $\varphi$ .
- El tubo  $T_1$  se obtiene de  $T_0$  seleccionando las valoraciones que hacen verdadera  $c_1$ . Para ello:
  - ★ Se eligen las que hacen verdadero el literal  $l_{1,1}$  (moléculas que “contienen” a  $l_{1,1}^1$ ).
  - ★ De las que hacen falso  $l_{1,1}$  (moléculas que **no** “contienen” a  $l_{1,1}^1$ ), se eligen las que hacen verdadero el literal  $l_{1,2}$  (moléculas que “contienen” a  $l_{1,2}^1$ ).
  - ★ De las que hacen falso  $l_{1,1} \vee l_{1,2}$ , se eligen las que hacen verdadero el literal  $l_{1,3}$ .
  - ★ Y así sucesivamente con todos los literales de  $c_1$ .
- El tubo  $T_2$  se obtiene de  $T_1$  seleccionando las valoraciones que hacen verdadera  $c_2$ . Para ello:
  - ★ Se eligen las que hacen verdadero el literal  $l_{2,1}$ .
  - ★ De las que hacen falso  $l_{2,1}$ , se eligen las que hacen verdadero el literal  $l_{2,2}$ .
  - ★ De las que hacen falso  $l_{2,1} \vee l_{2,2}$ , se eligen las que hacen verdadero el literal  $l_{2,3}$ .
  - ★ Y así sucesivamente con todos los literales de  $c_2$ .
- El proceso se reitera  $p$  veces hasta obtener el **tubo de salida**  $T_p$ , a partir de  $T_{p-1}$ .

# Experimento de Lipton

Primer esquema algorítmico molecular.

**Entrada:**  $T_0$

**para**  $i \leftarrow 1$  **hasta**  $p$  **hacer**

$T \leftarrow \emptyset$

**para**  $j \leftarrow 1$  **hasta**  $r_i$  **hacer**

$T' \leftarrow + (T_{i-1}, l_{i,j}^1)$

$T_{i-1} \leftarrow - (T_{i-1}, l_{i,j}^1)$

$T \leftarrow T \cup T'$

$T_i \leftarrow T$

**Detectar** ( $T_p$ )

# El problema 3-COL

Sea  $G = (V, E)$  un grafo no dirigido.

- \* Una **coloración** de  $G$  con  $k$  colores es una aplicación  $f$  de  $V$  en  $\{1, \dots, k\}$ .
- \* Una **coloración**  $f$  de  $G$  es **válida** si para cada  $x, y \in V$  se verifica que  $\{x, y\} \in E \Rightarrow f(x) \neq f(y)$ .
- \* Diremos que  $G$  es **coloreable** con  $k$  colores si existe una coloración válida de  $G$  con  $k$  colores.

Enunciado del problema **3-COL**: **Dado un grafo no dirigido  $G$ , determinar si existe una coloración válida de  $G$  con tres colores.**

- \* Se prueba fácilmente que el problema **3-COL** pertenece a la clase **NP**.

**Teorema:** *El problema 3-COL es NP-completo (presuntamente intratable).*

# El problema 3-COL

Un programa molecular que resuelve “**eficientemente**” el problema 3-COL:

**Entrada:**  $T$  (contiene todas las posibles coloraciones de  $G$  con 3 colores)

para  $i \leftarrow 1$  hasta  $p$  hacer

$$T_1 \leftarrow +(T, e_i^1(c_1)); T_1^* \leftarrow -(T, e_i^1(c_1))$$

$$T_2 \leftarrow +(T_1^*, e_i^1(c_2)); T_3 \leftarrow -(T_1^*, e_i^1(c_2))$$

para  $j \leftarrow 1$  hasta 3 hacer

$$T'_j \leftarrow -(T_j, e_i^2(c_j))$$

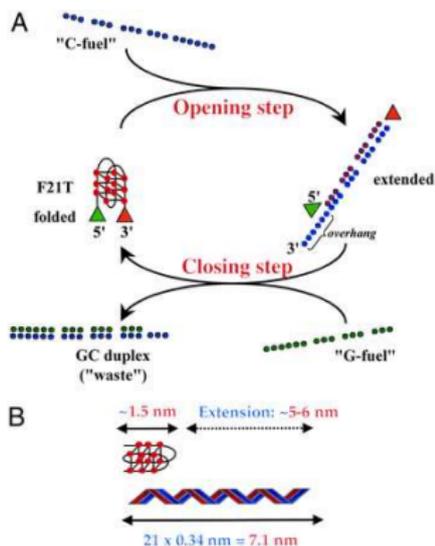
$$T \leftarrow T_1' \cup T_2'$$

$$T \leftarrow T \cup T_3'$$

**Detectar** ( $T$ )

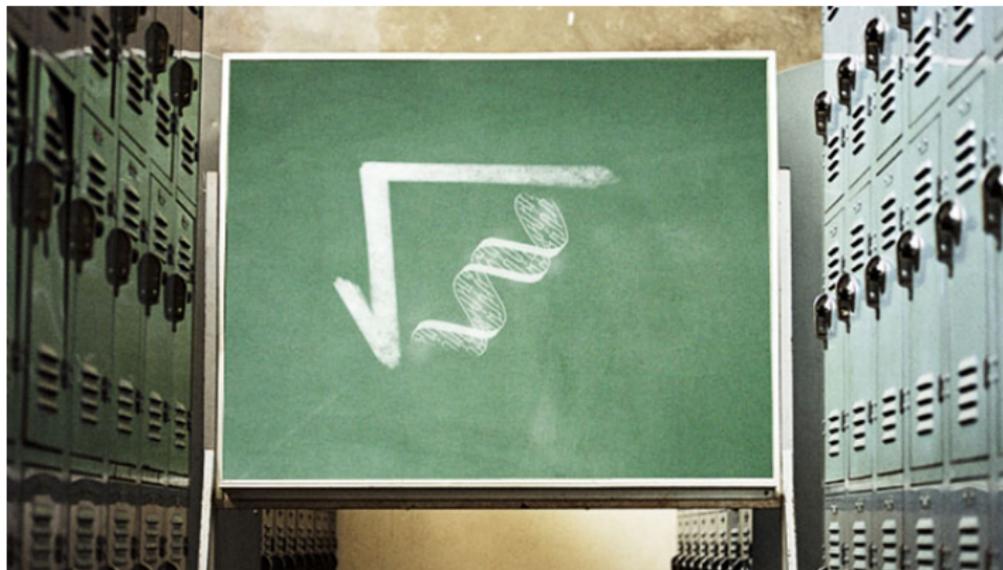
# Máquina de ADN que simula movimientos musculares

Esquema de una máquina molecular basada en ADN que realiza movimientos de extensión y contracción.



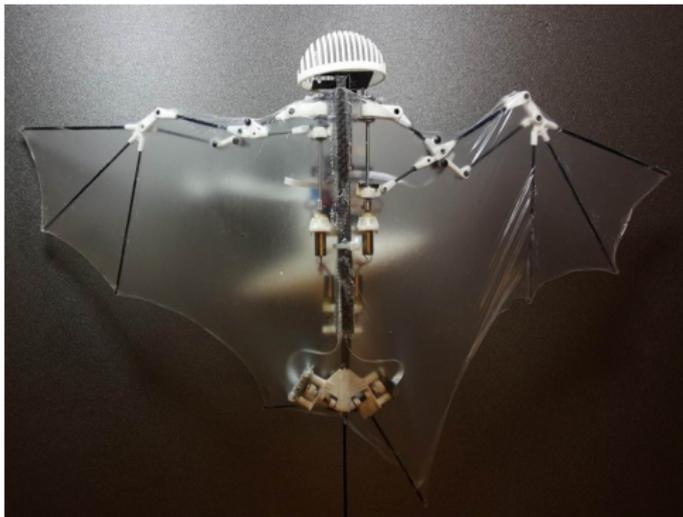
(P. Alberti, J.L. Mergny, 2003)

# Máquina de ADN que calcula raíces cuadradas



(Lulu Qian y Erik Winfree, junio 2011)

- La máquina está formada por 130 moléculas de ADN.
- Ha calculado las raíces cuadradas de 0,1,4 y 9 (tarda unas 4 horas en realizar el cálculo).

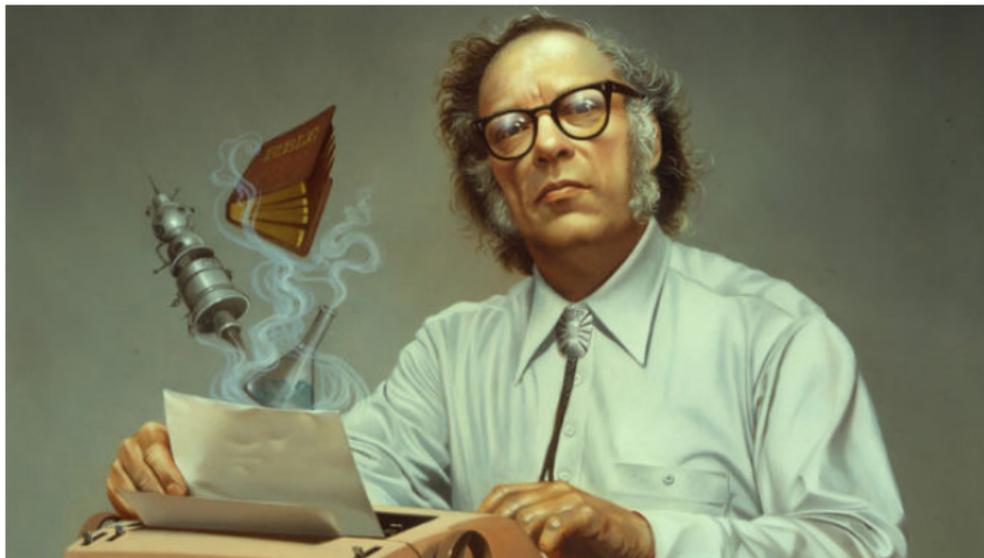


# Robótica

Término **robot**: proviene de la palabra checa **robota** (trabajos forzados, servidumbre).

Aparece por primera vez en una novela de Karel Čapek (1920): **Rossum's Universal Robots**.

**Robótica**: ciencia que estudia los robots (acuñado por I. Asimov, 1942).



# Robótica

Disciplina relevante de la IA:

- \* Construcción de máquinas que simulan **comportamientos inteligentes**.

**Robot:** máquina que de manera autónoma,

- \* Percibe su entorno.
- \* Toma decisiones.
- \* Actúa en consecuencia.

**Robot humanoide:** máquina que trata de simular el **comportamiento humano**.

# Robótica



¿Cómo conseguir que una máquina simule un proceso inteligente?

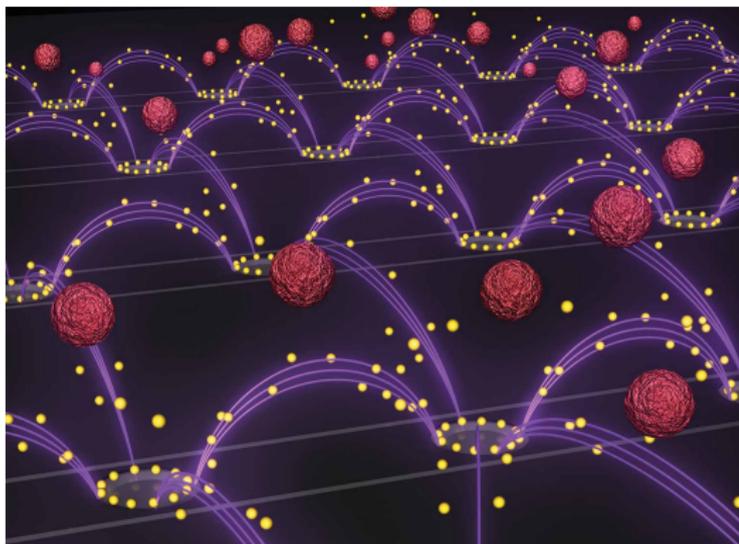
- ★ Es necesario **formalizar** dicho proceso (usando teorías matemáticas).

# Nanocomputación

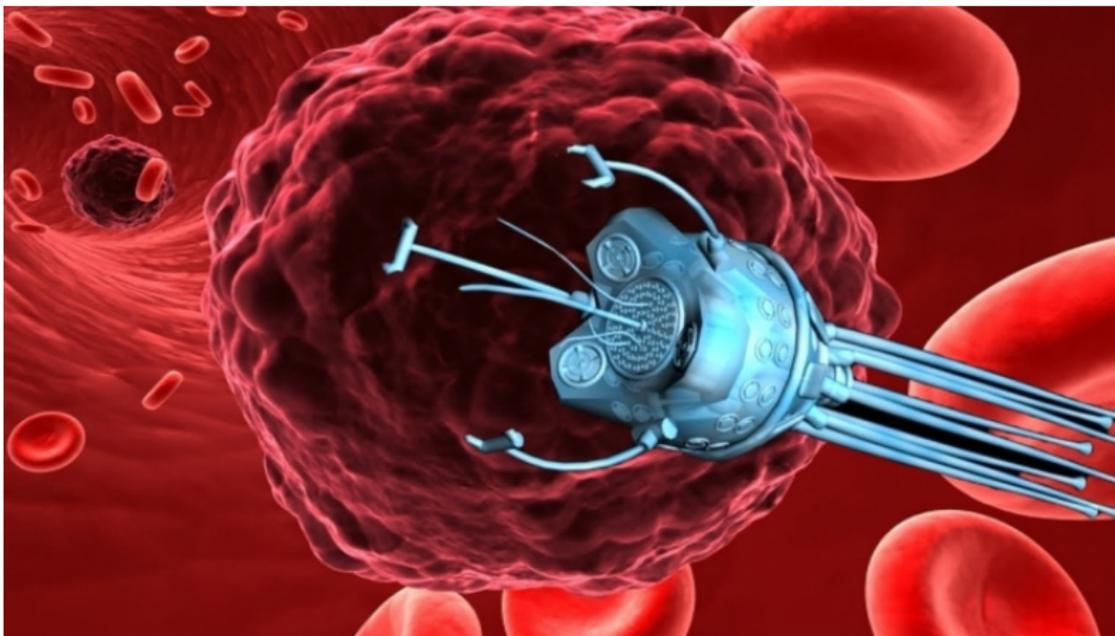
**Escala nanométrica:**  $1\text{nm} = 10^{-9}\text{ mm}$  (1.000 millones de nm es igual a 1 mm)

Nanocomputación: dispositivos computacionales a escala nanométrica.

Precursor de la nanocomputación: **Richard Feynman** (1959).



# Nanorobots



Robots a escala nanométrica ( $10^9$  nm = 1 mm).

# Nanorobots moleculares basados en ADN

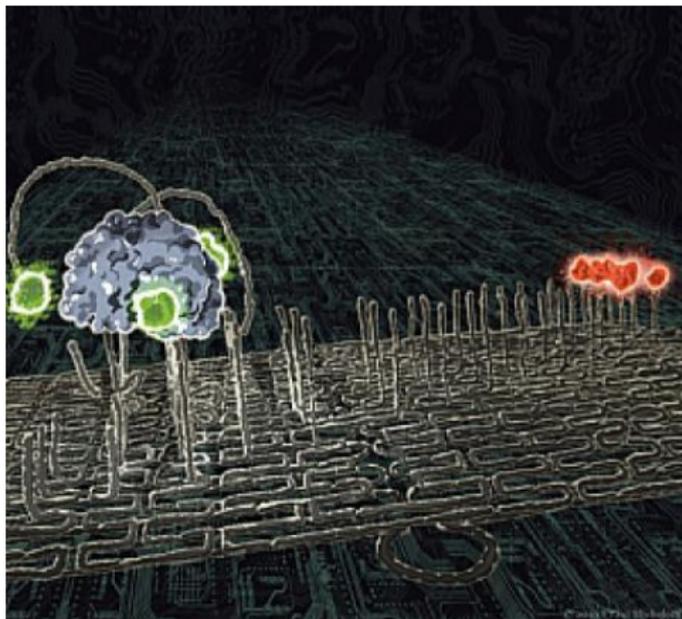
Obreros en una cadena de montaje (Erik Winfree, 2010)



Más de mil nanorobots se comunican entre sí y trabajan coordinadamente (U. Harvard, 2014).

# Nanorobots moleculares basados en ADN

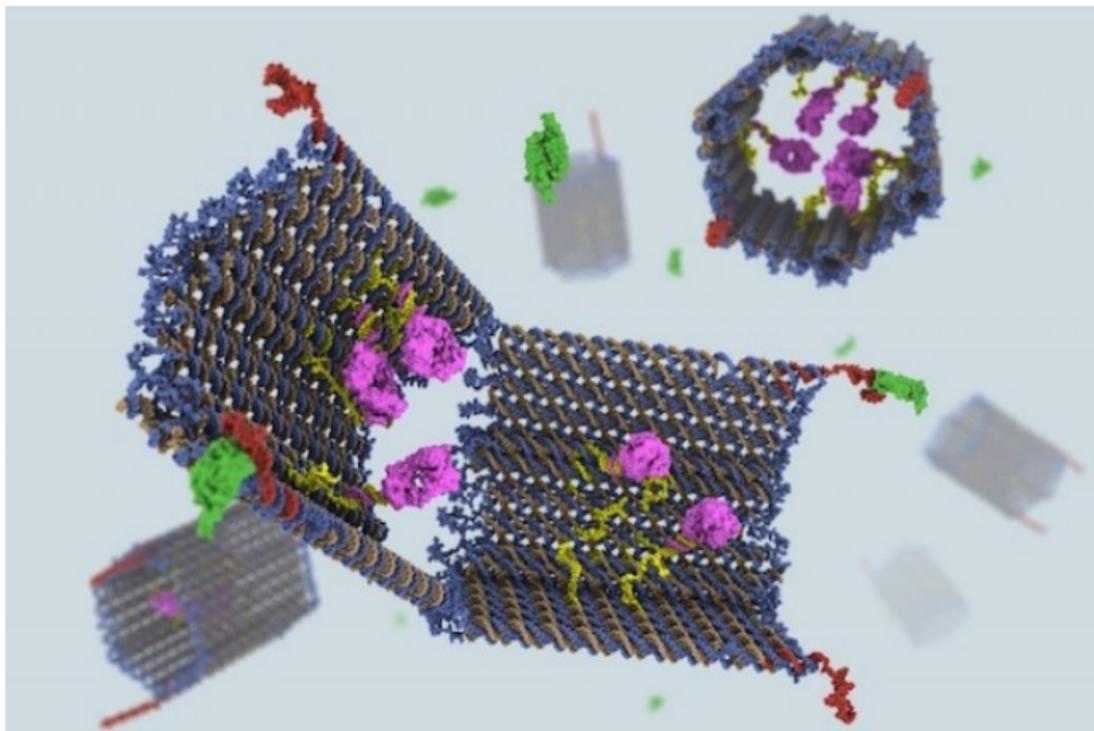
Una **araña molecular** basada en ADN (Lloyd Smith, 2010)



Es una proteína unida a cuatro “patas” (hebras de ADN). Se desplaza sobre un origami de ADN.

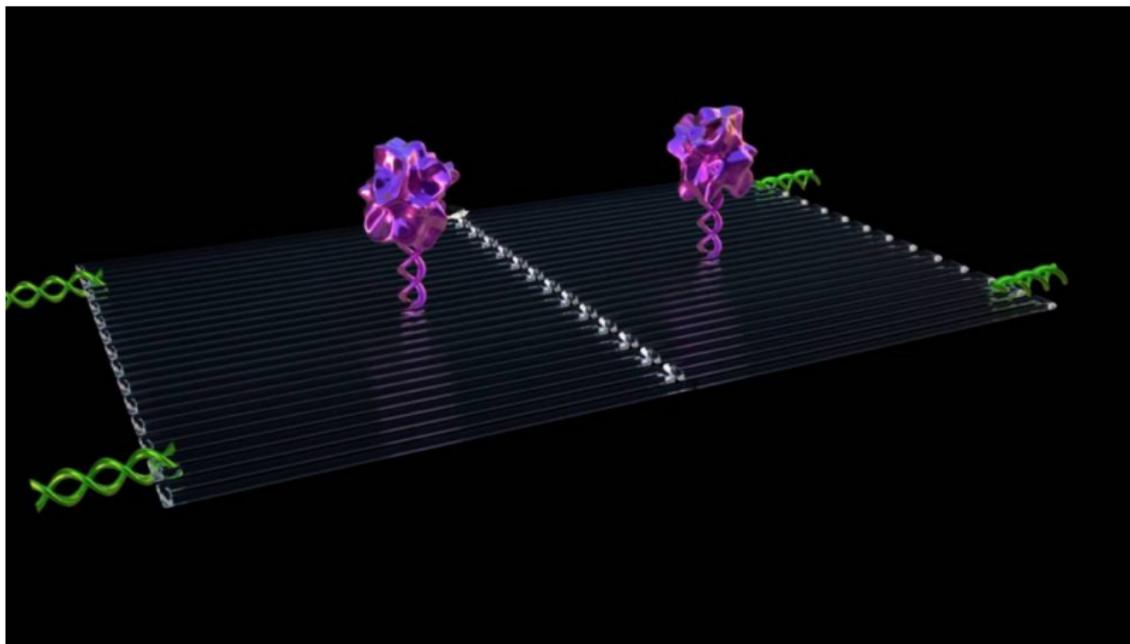
Ha conseguido dar 50 pasos, unos 100 nm (hasta entonces sólo se había conseguido caminar 3 pasos).

# Nanorobots moleculares basados en ADN



Reconoce células tumorales y les inyecta directamente los medicamentos para destruirlas (Harvard's Wyss Institute, 2012 y Davis Cancer Centre de la Universidad de California, 2014).

# Nanorobots moleculares basados en ADN



Se inyectan en los vasos sanguíneos que llevan sangre al tumor (Universidades del estado de Arizona y el Centro Nacional de Nanociencia y Tecnología de China: 2018).