andy potts

Life’s code script

Turing machines and cells have much in common, argues Sydney Brenner

http://www.nature.com/news/specials/turing/

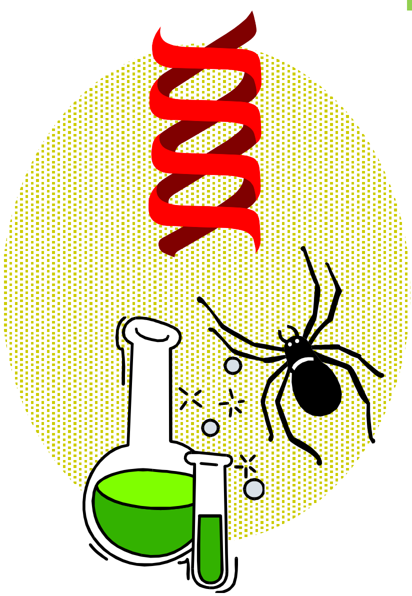
natural computation

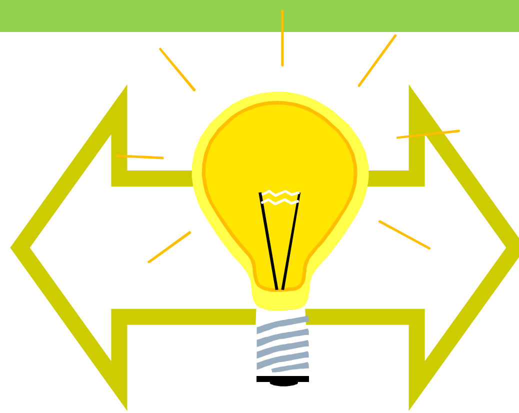
dna sorting &
networks

understanding nature
as a computational process

neural netw &
genetic alg

bio inspired computing





bio hardware

this talk

bio-informatics

many facets of …

Lila Kari, Grzegorz Rozenberg:
The many facets of natural computing.
CACM 51 (8) 72-83, okt 2008

Turing’s Legacy

SNiC – Utrecht

7 maart 2012

Computer in a TestTube

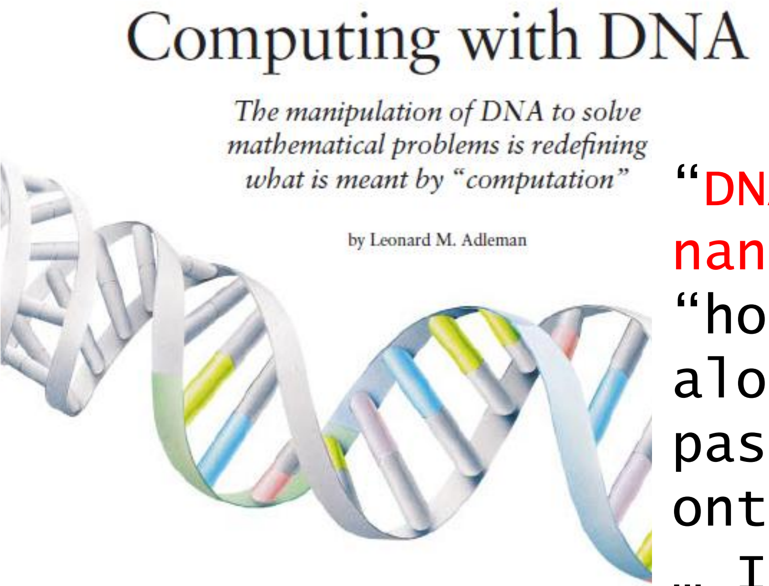
molecular computing

Hendrik Jan Hoogeboom
Computer Science Leiden

Len Adleman

Molecular Computation of Solutions to Combinatorial Problem, Science, 266: 1021-1024, (Nov. 11) 1994.

http://www.usc.edu/dept/molecular-science/fm-adleman.htm



Scientific American

"DNA polymerase is an amazing little nanomachine, a single molecule that "hops" onto a strand of DNA and slides along it, "reading" each base it passes and "writing" its complement onto a new, growing DNA strand … I was struck by its similarity to something described in 1936 by Alan M. Turing, the famous British mathematician …

This realization caused me to sit up in bed and remark to my wife, Lori, 'Jeez, these things could compute.' I did not sleep the rest of the night, trying to figure out a way to get DNA to solve problems."

Leonard M. Adleman - Computing with DNA
Scientific American August 1998

Physicists plunder life's tool chest

If we look inside the cell, we see extraordinary machines that we couldn't make ourselves, says Len Adleman. “It's a great tool chest - and we want to see what can we build with it.”

‘travelling salesman’ problem

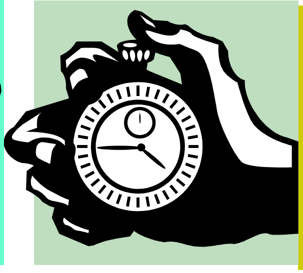
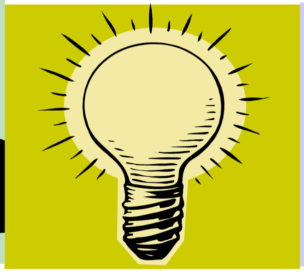
Nature News Service 2003

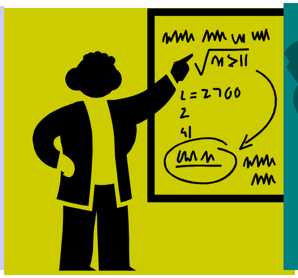
DNA excels at getting an astronomical amount of data into a tiny space. “One gram of DNA can store as much information as a trillion compact discs,” says Adleman. Myriad DNA molecules can examine every possible route at once, rather than one at a time, as in a conventional computer.

massive parallellism

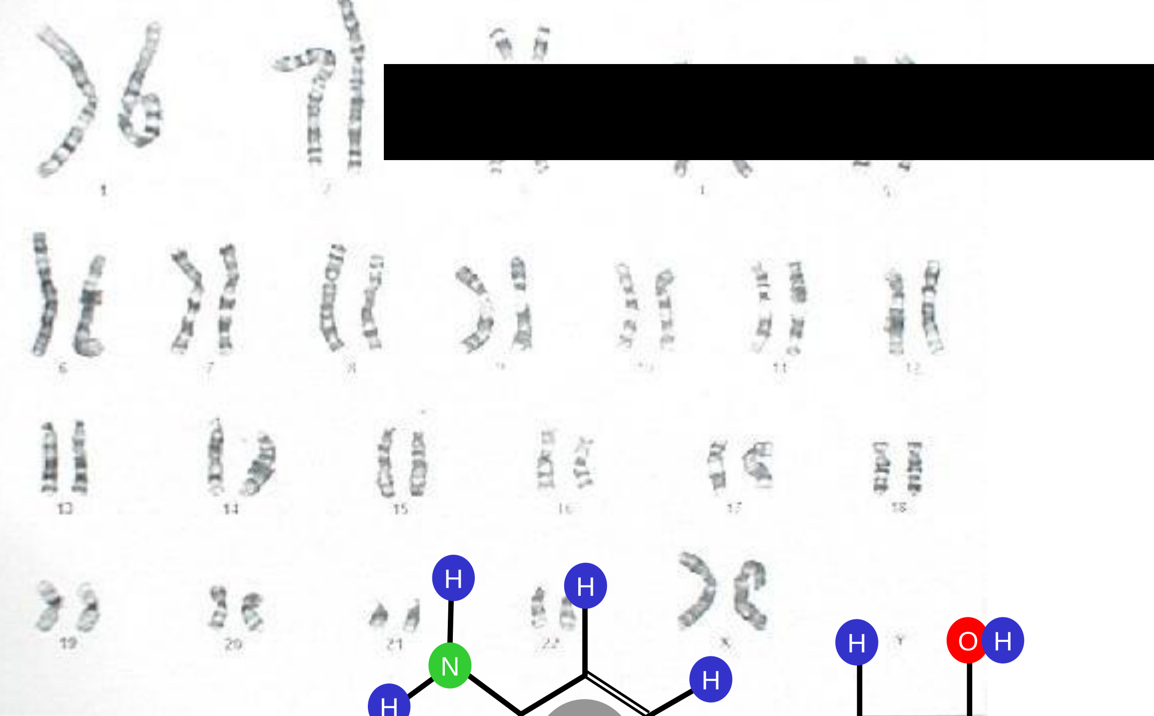
contents

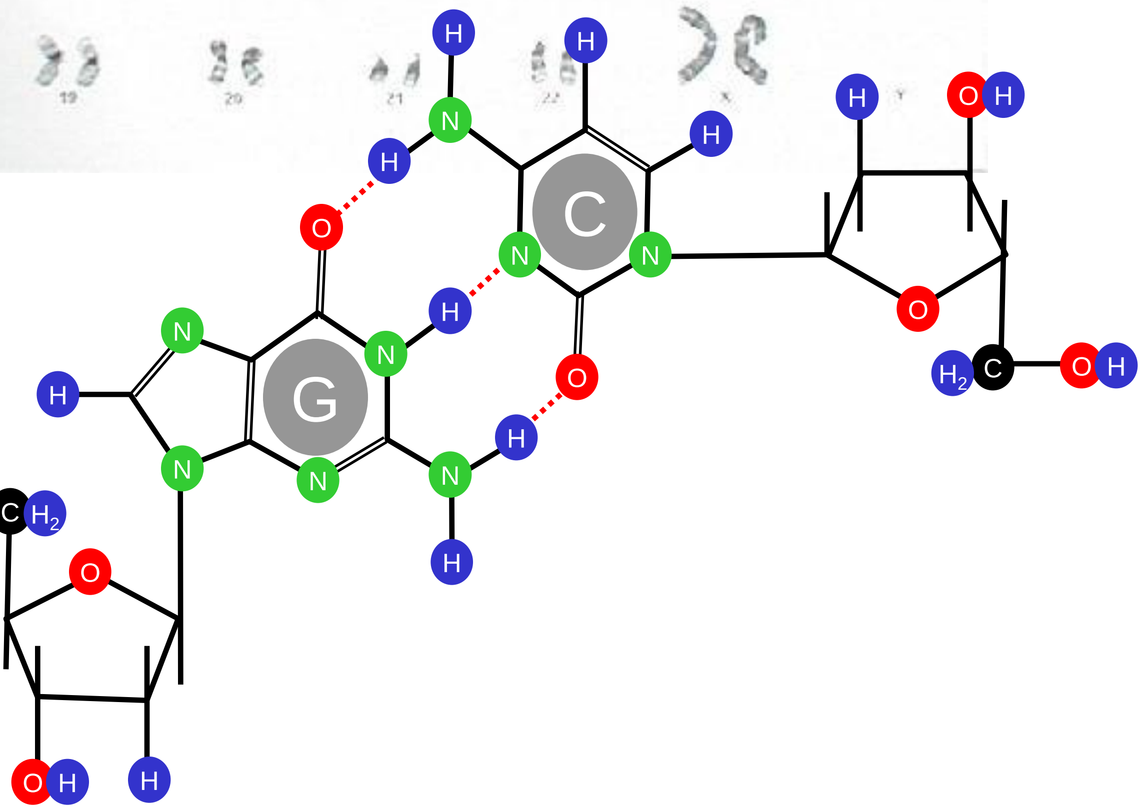
- DNA … the tool chest
- Hamilton Path Problem
- Adleman's algorithm
- comments
- theory … Turing machine
- recent work + future
- self assembly

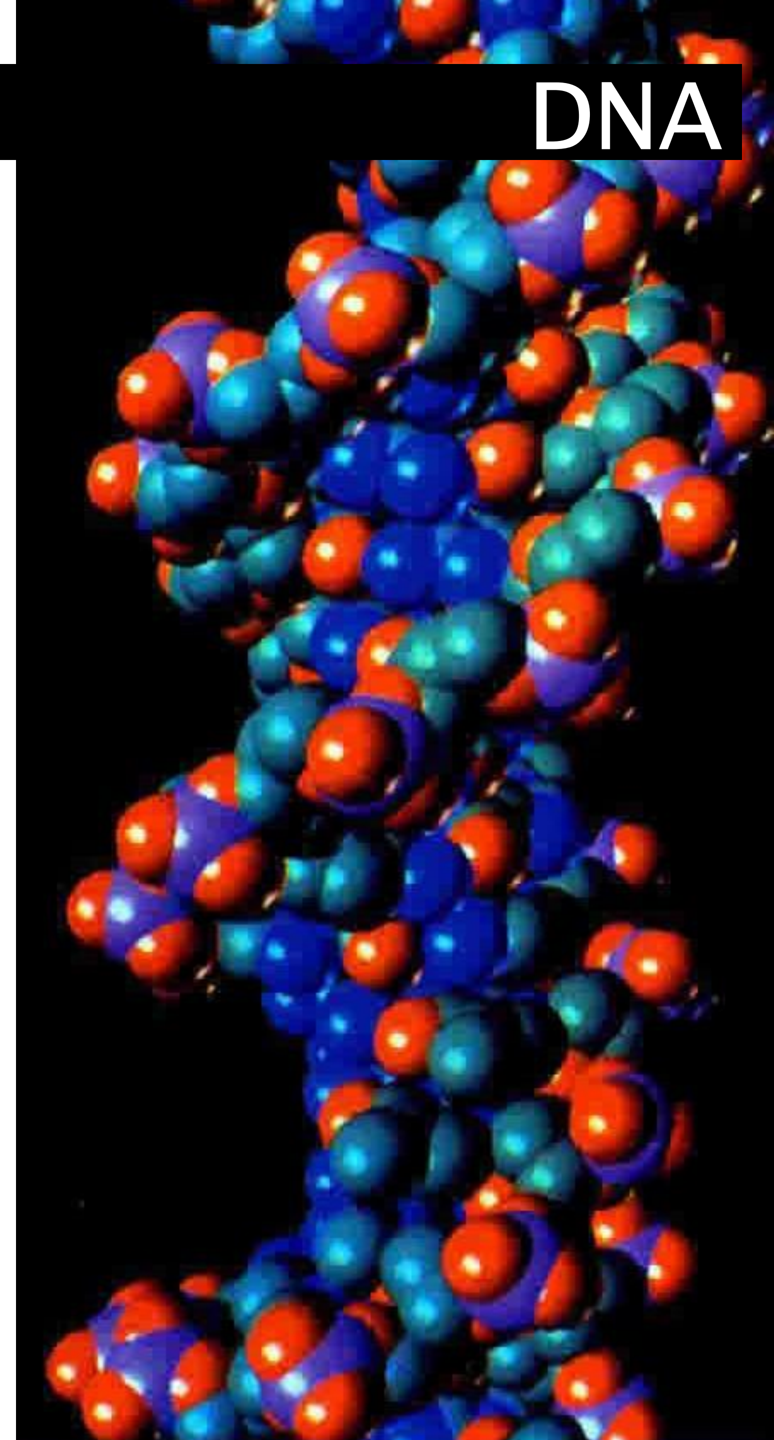






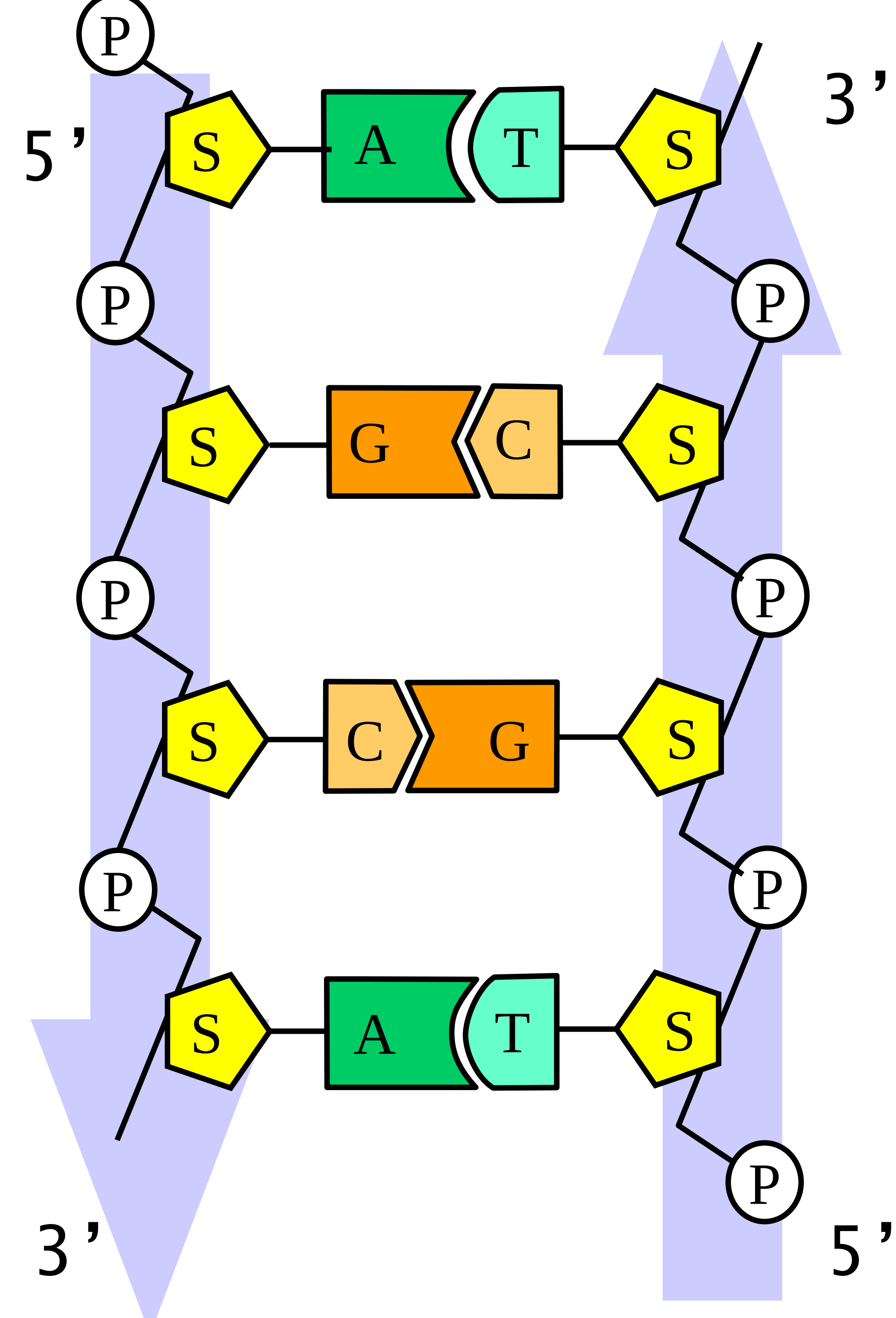
DNA







DNA



Base pairs

Watson & Crick

[& Rosalind Franklin]

A=T

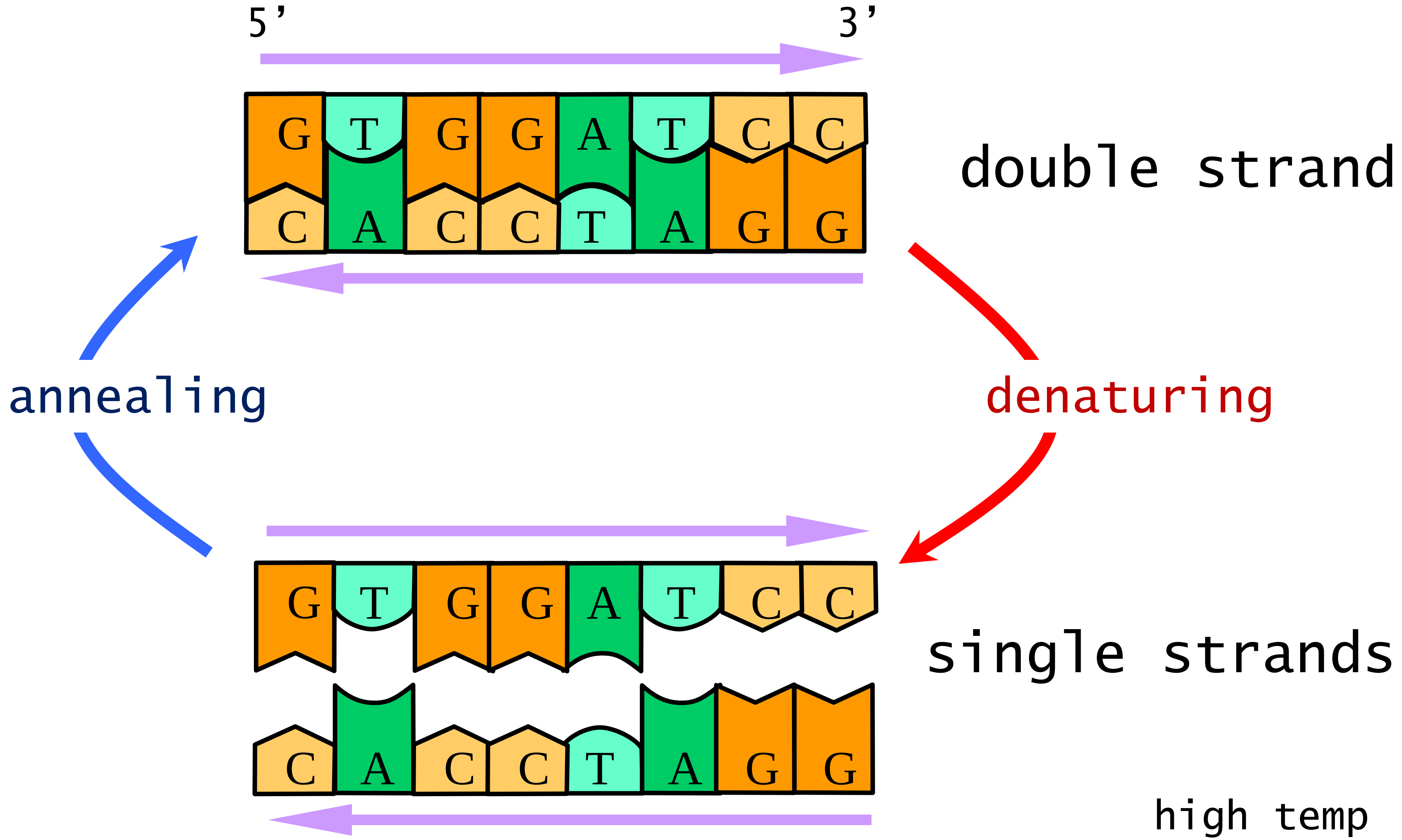
adenine - thymine

C≡G

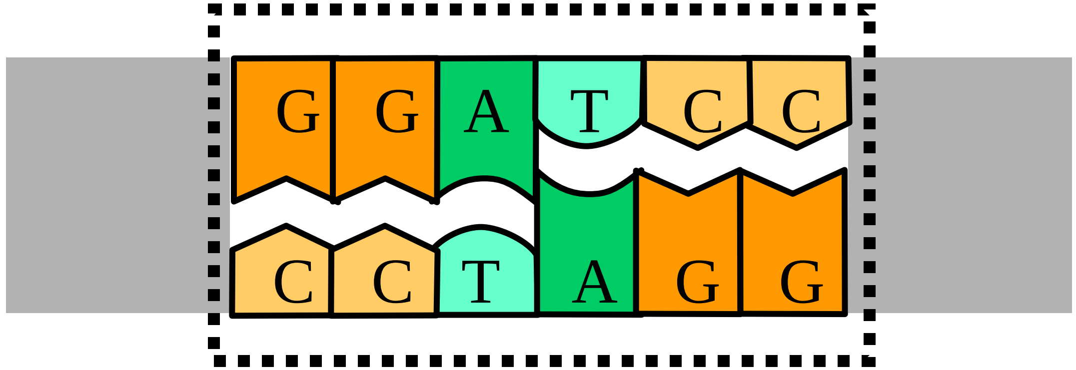
guanine - cytosine

single – double strand

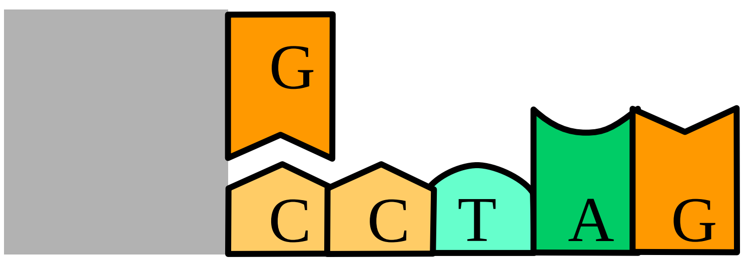
‘complementarity’

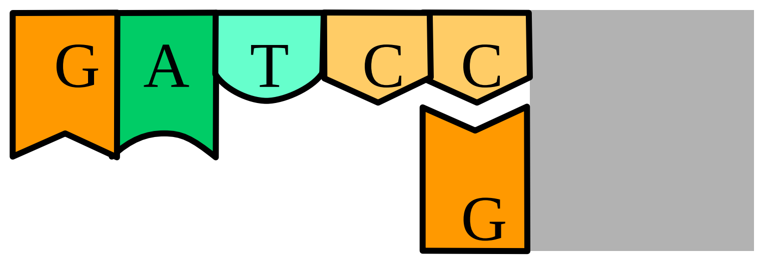


restriction enzymes



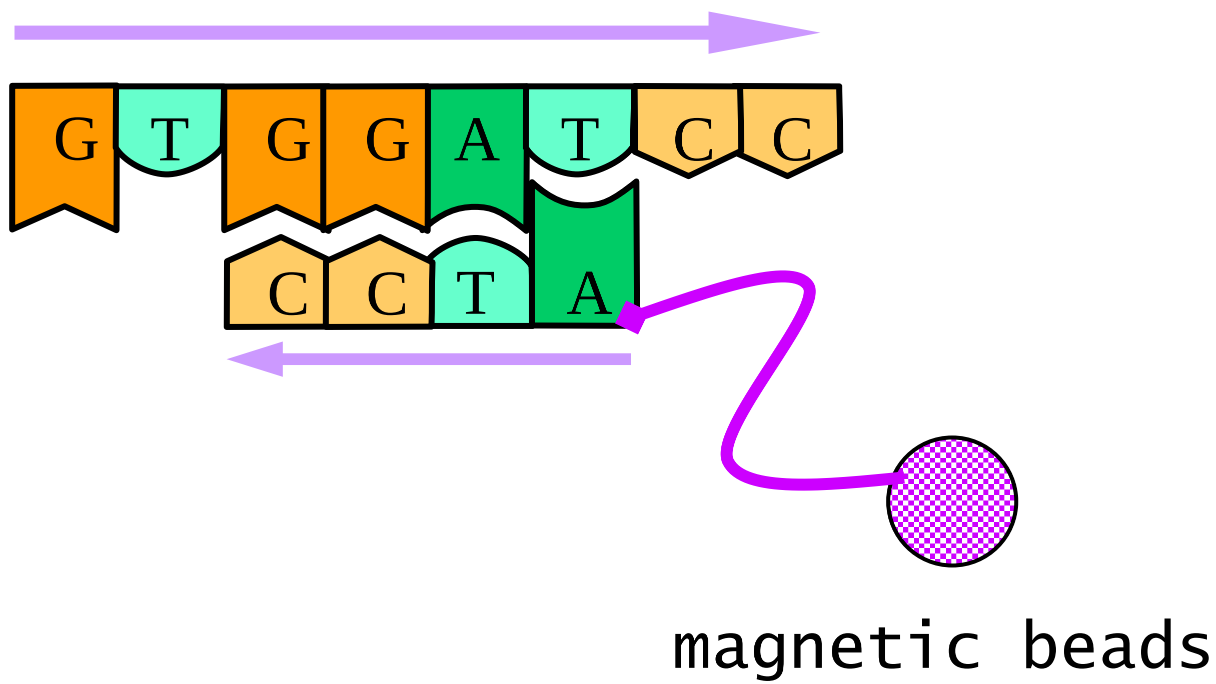
BamHI





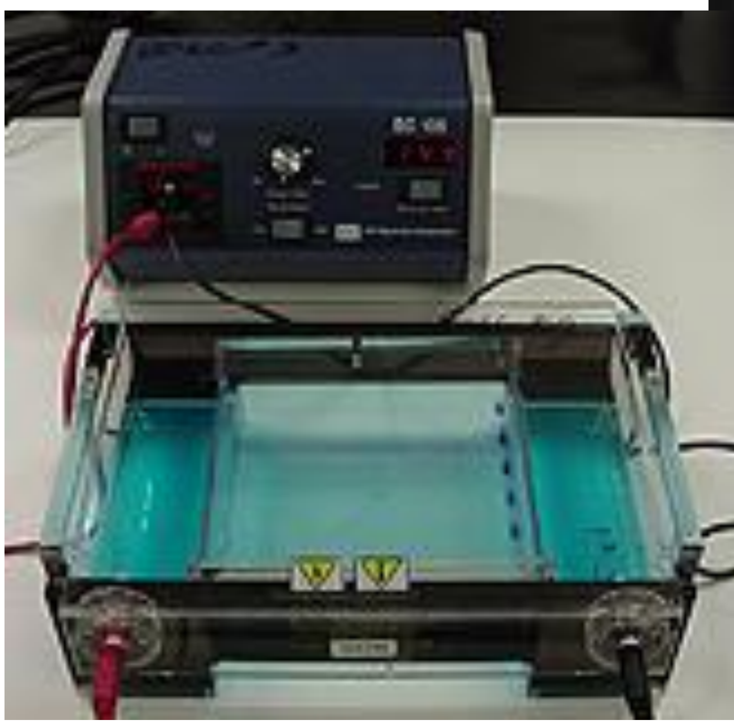
sticky ends

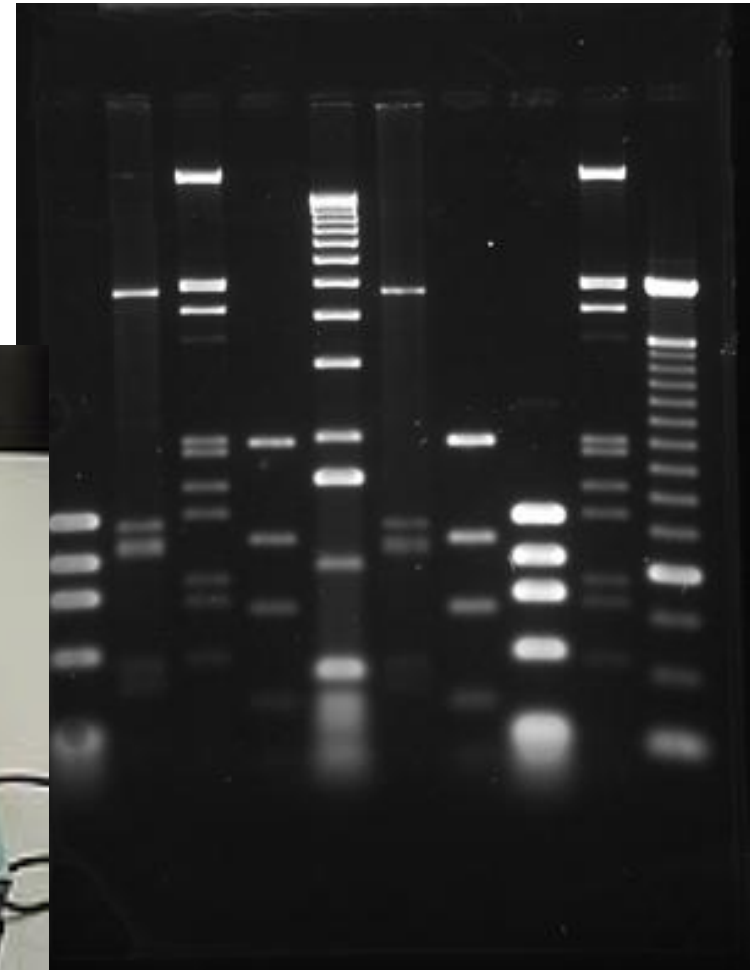
subsequence selection



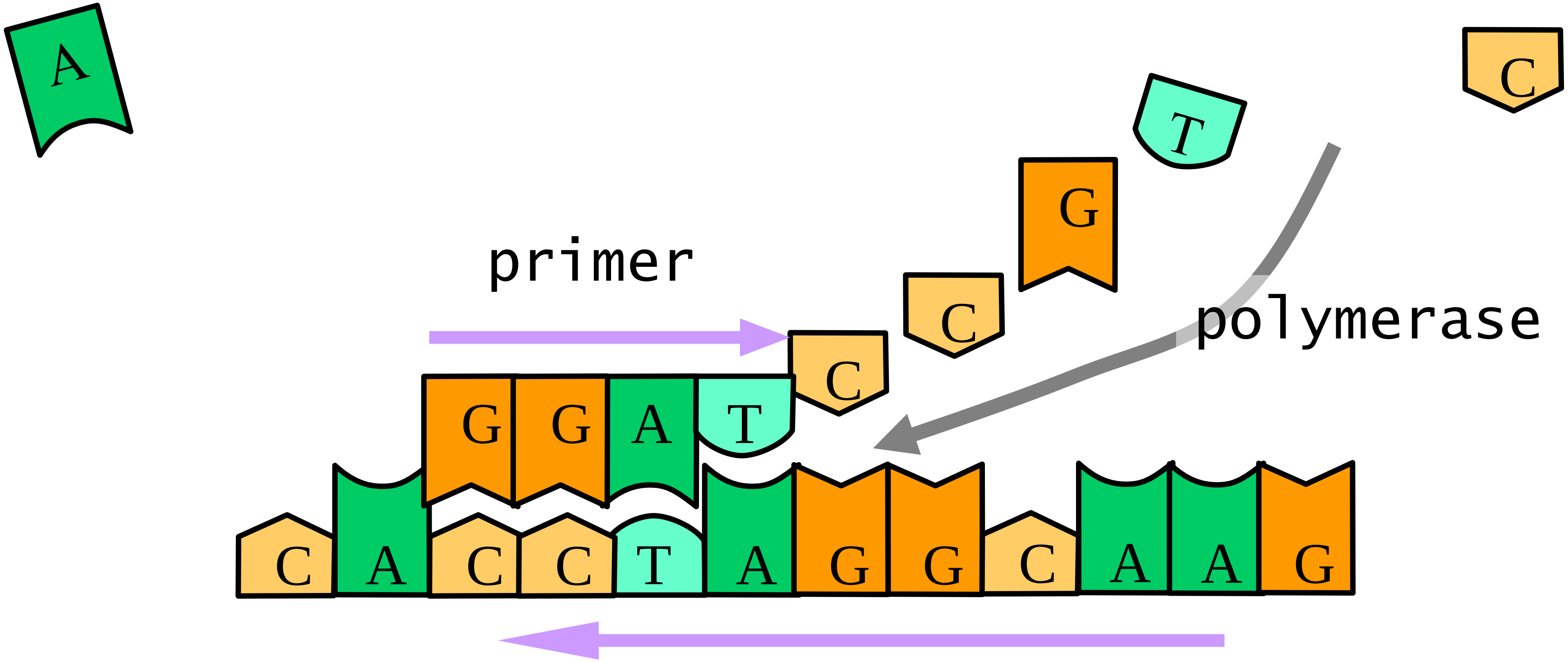
separation on length

DNA gel electrophoresis





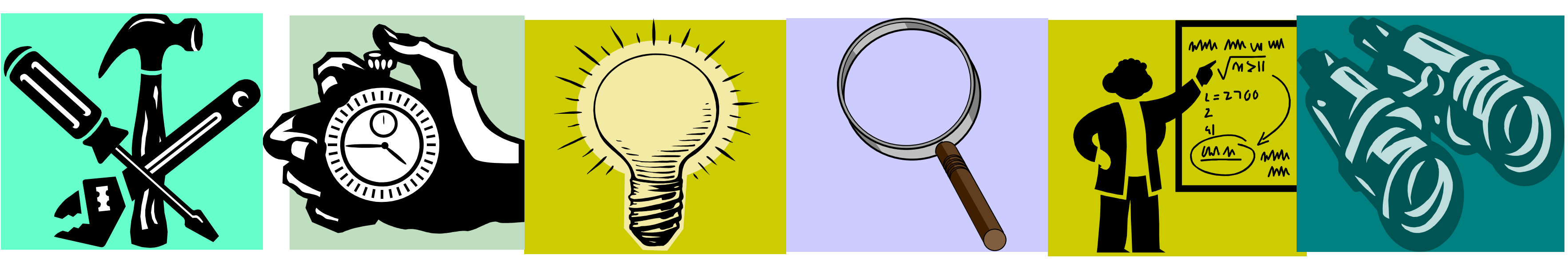
multiplication / amplification



PCR – polymerase chain reaction

contents

- DNA … the tool chest
- Hamilton Path Problem
- Adleman's algorithm
- comments
- theory … Turing machine
- recent work + future
- self assembly



The general idea

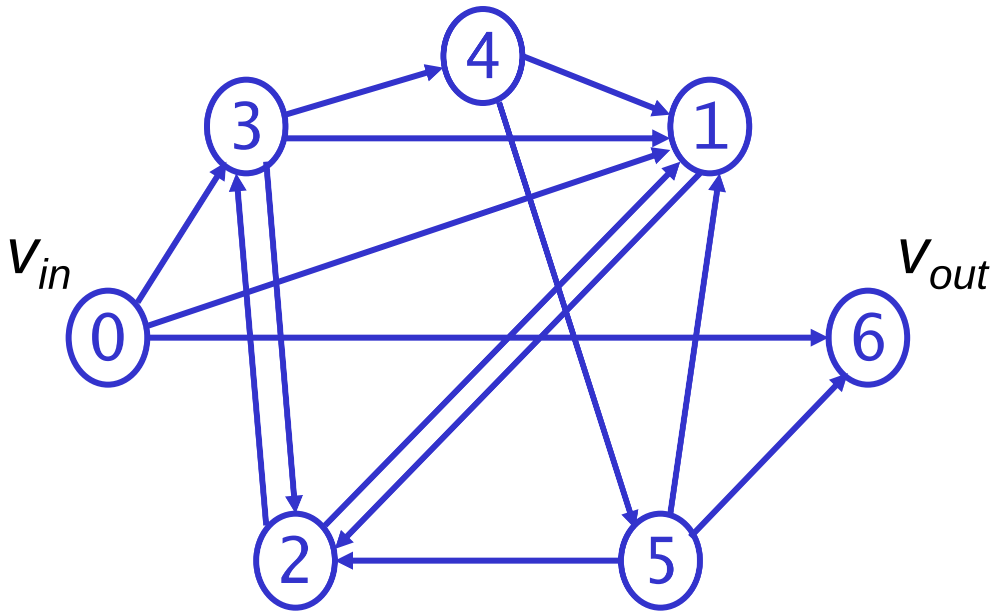
custom made
single strands of DNA
(many copies)

is there a
double strand
with my desired
properties?

properties:
- length,
- subsequence.

if we can do this, then we can solve
certain problems (efficiently)!

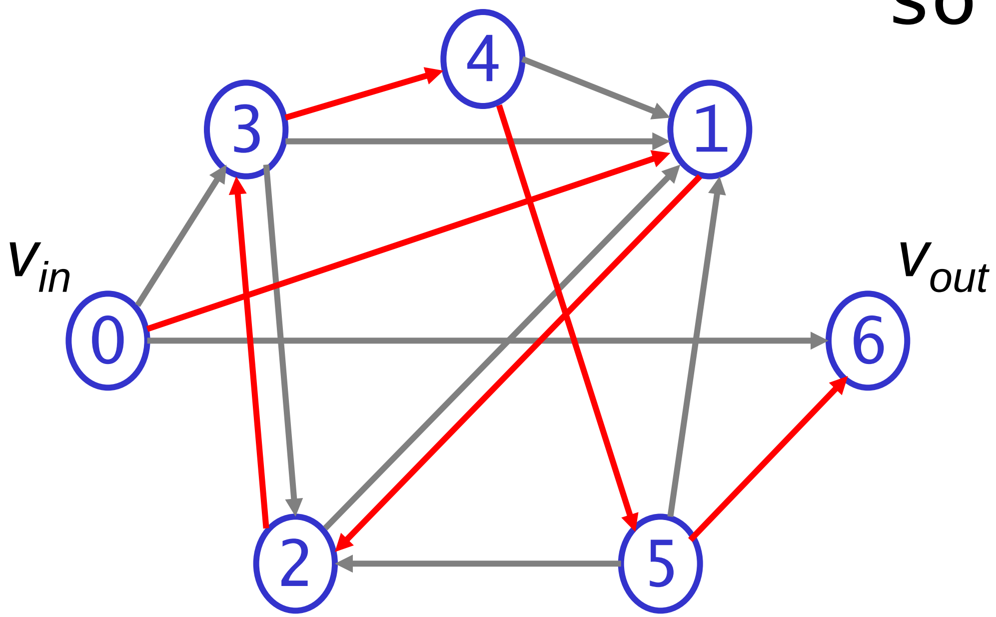
HPP: Hamilton Path Problem



'travelling salesman'

given: directed graph (points & connections)
question: is there a path that visits each point exactly once ?

HPP: Hamilton Path Problem

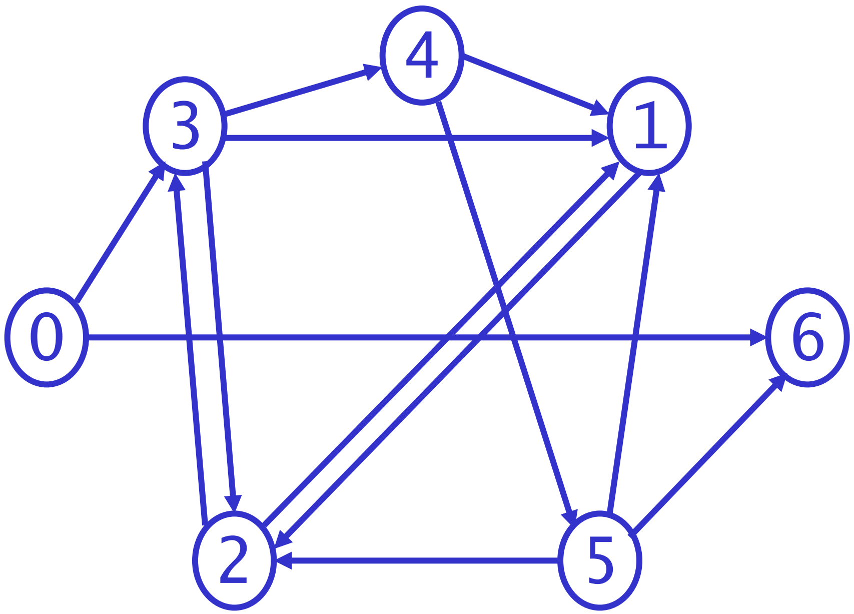


solution

'travelling salesman'

given: directed graph (points & connections)
question: is there a path that visits each point exactly once ?

HPP: Hamilton Path Problem

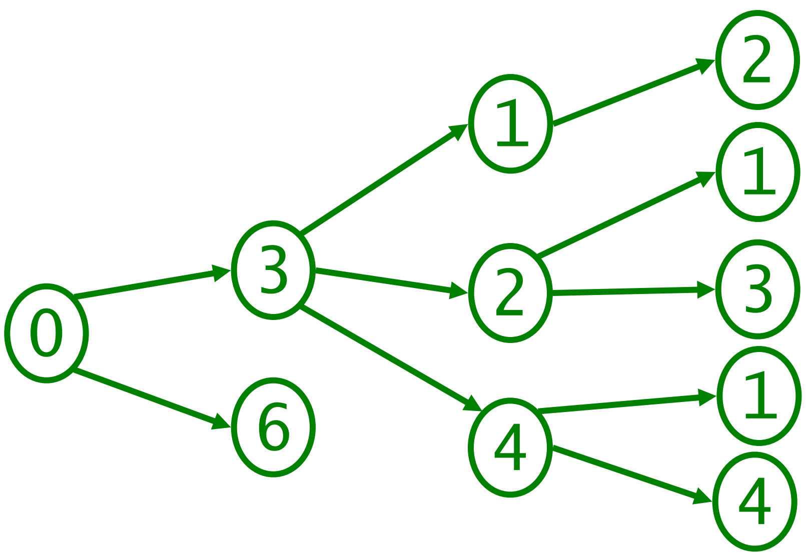


no solution?

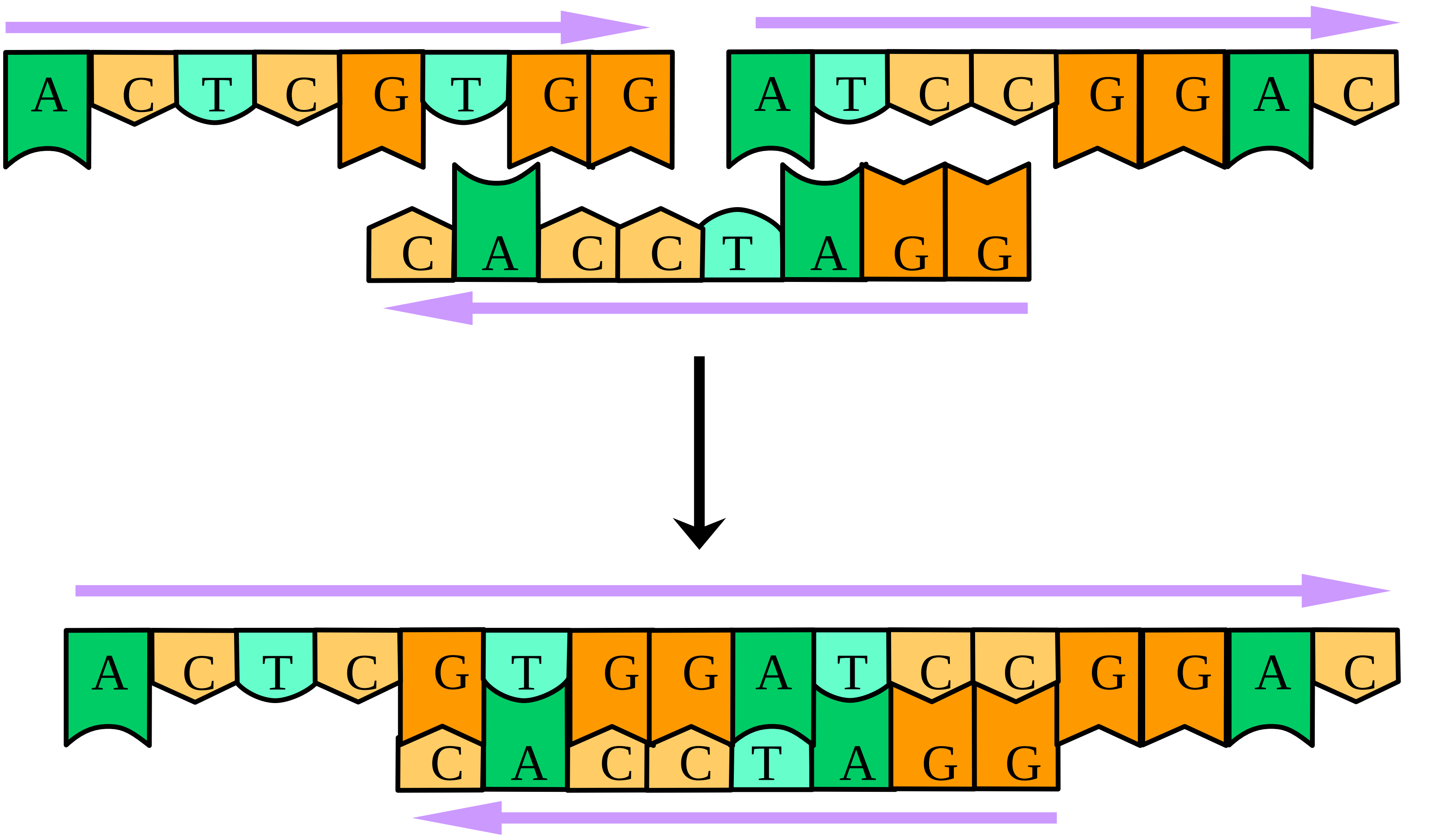
exponential time:
try all possibilities

representative class
'**NP complete**'

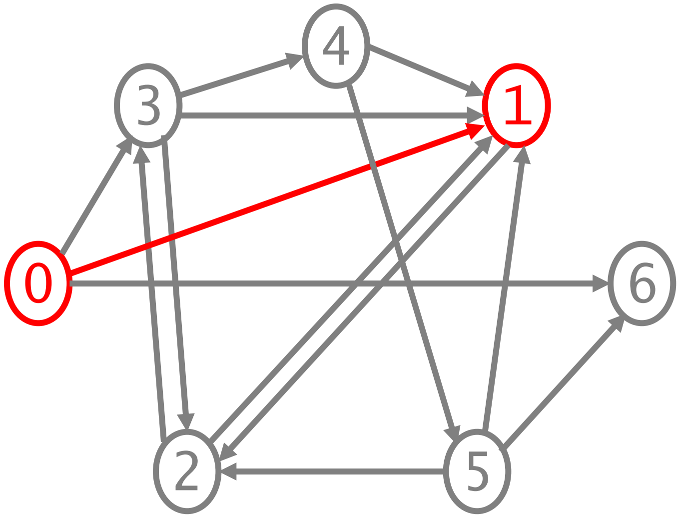
heuristics



building blocks



Adleman's algorithm



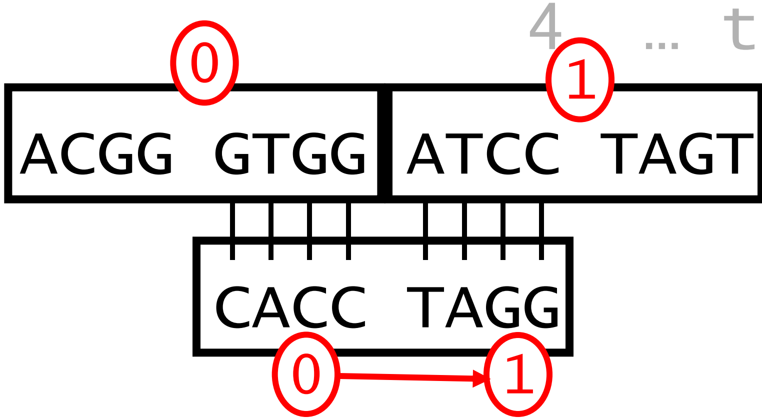
0. coding the graph

1. generate 'all' paths

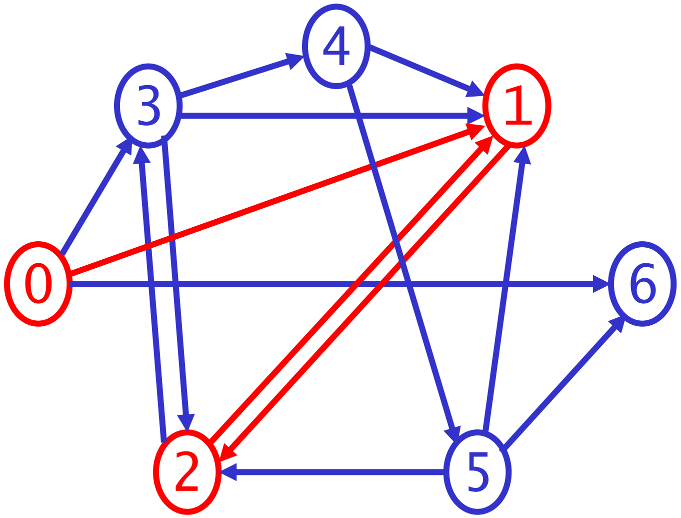
keep only paths

2. … from v_{in} to v_{out}
3. … that enter n vertices
4. … that enter all vertices

any path remains OK



Adleman's algorithm



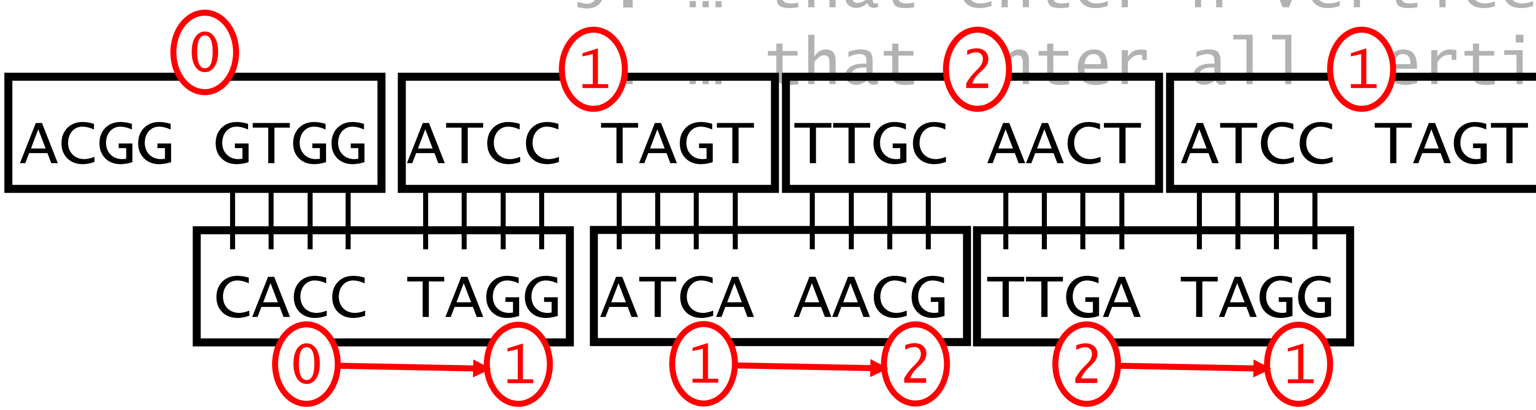
0. coding the graph

1. generate 'all' paths

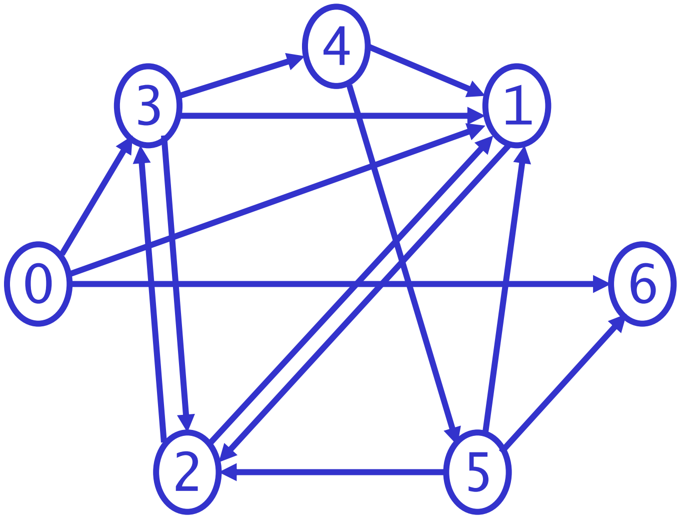
keep only paths

2. … from v_{in} to v_{out}
3. … that enter n vertices

… that enter all vertices



Adleman's algorithm



0. coding the graph
1. generate 'all' paths

   keep only paths
2. … from v_{in} to v_{out}
3. … that enter n vertices
4. … that enter all vertices

5. if any path remains OK

- PCR with v_{in} and v_{out} primers
- gel: separate on length, amplify & purify
- magnetic beads: select strands
- PCR amplification & gel

comments …

- “clear that the methods could be scaled up to … larger graphs”

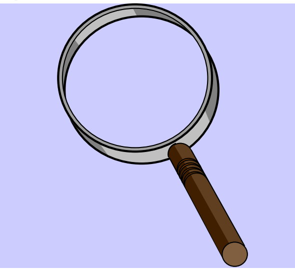
    + bath tub of DNA ?
    + suitable algorithms

- approximately 7 days of lab work

    + automation
    + alternative molecular algorithms

- possibility of errors

    + pseudopaths: accidental ligation
    + PCR, separation procedures
    + hairpin loops
    + stability when scaled



comments …

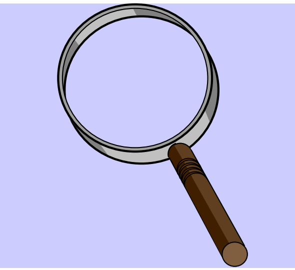
- "power of this method of computation"
  - 10^{14} operations 10^{20} plausable
  - exceed supercomputers by thousandfold

  :)

- "not clear whether … used to solve real computational problems"

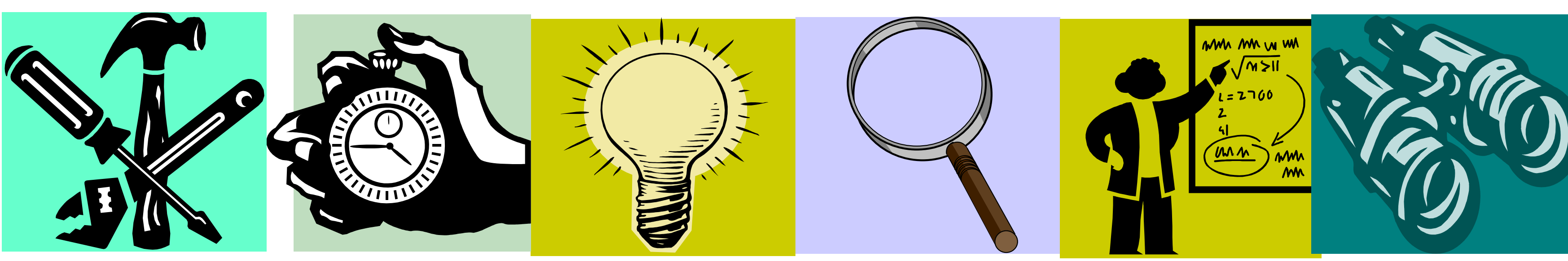
  . multiplying 100 digit numbers

- potential: massively parallel searches

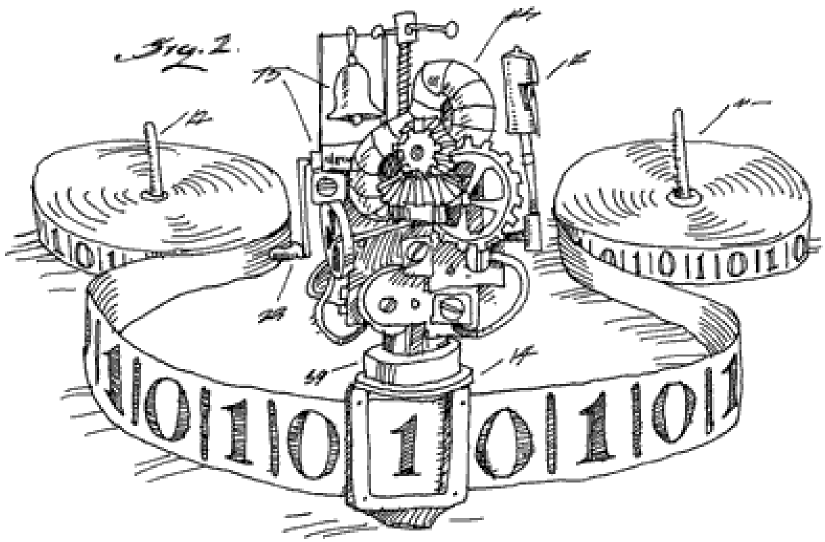


contents

- DNA … the tool chest
- Hamilton Path Problem
- Adleman's algorithm
- comments
- theory … Turing machine
- recent work + future
- self assembly



Turing machine



GGATGnnnnnnnnn
CCTACnnnnnnnnnnnnn

Turing Machine by Tom Dunne
American Scientist, March-April 2002

FokI

Catalog #	Size	Concentration	Price	Qt
R0109L	5,000 units	4,000 units/ml	$244.00	1
R0109S	1,000 units	4,000 units/ml	$61.00	1

Prices are in US dollars and valid only for US orders.

Download: MSDS PDF

Recognition Site:

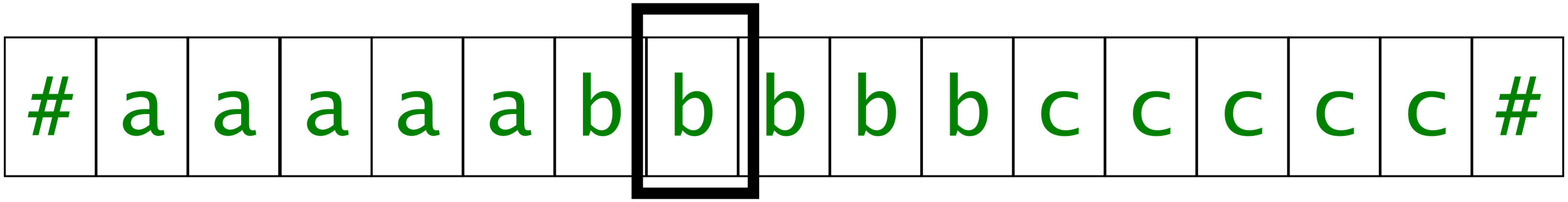
5′. . . G G A T G(N)$_9$▼. . . 3′
3′. . . C C T A C(N)$_{13}$▲. . . 5′

isoschizomers | compatible ends | single letter code

Source:
A *E. coli* strain that carries the FokI gene from *Flavobacterium okeanokoites*

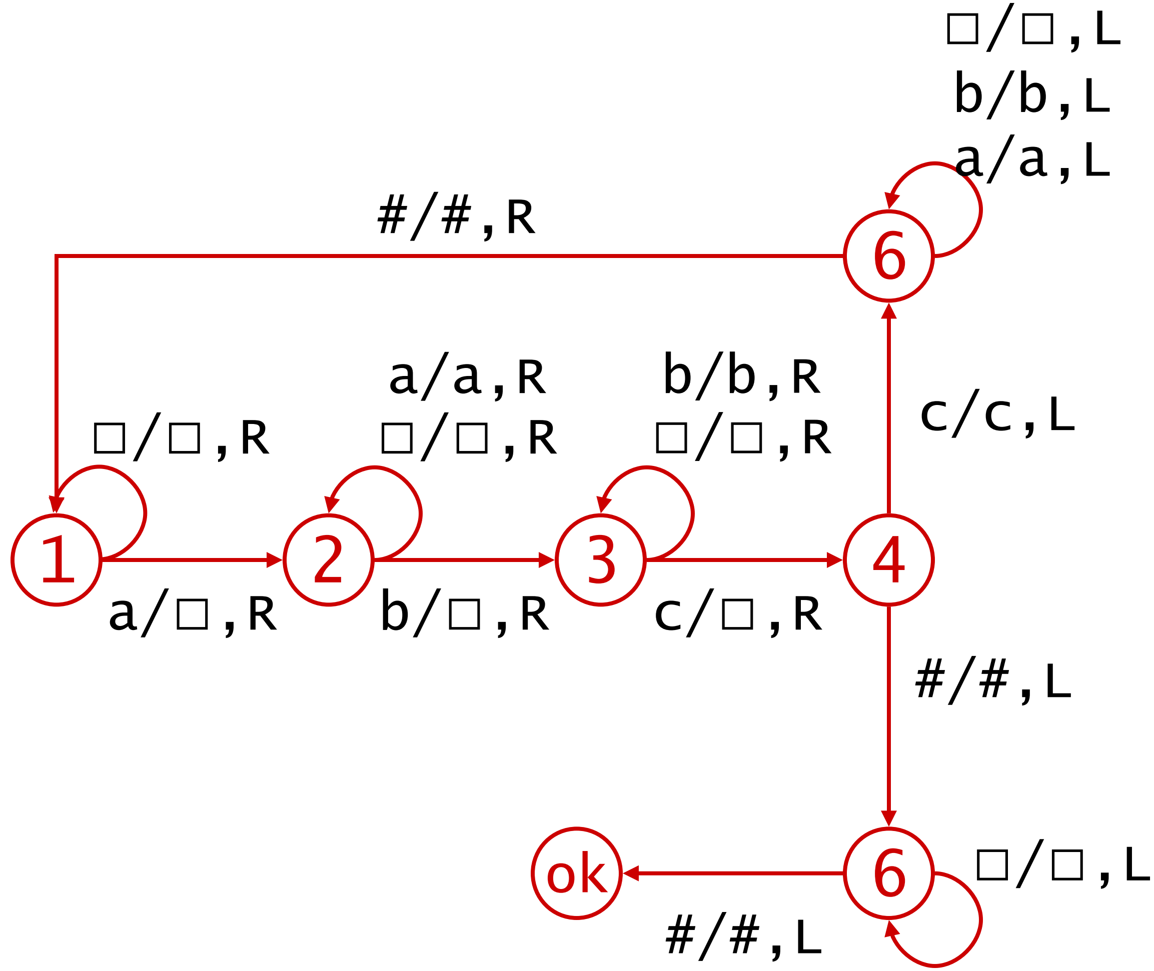
Reagents Supplied:
NEBuffer 4

Turing machine

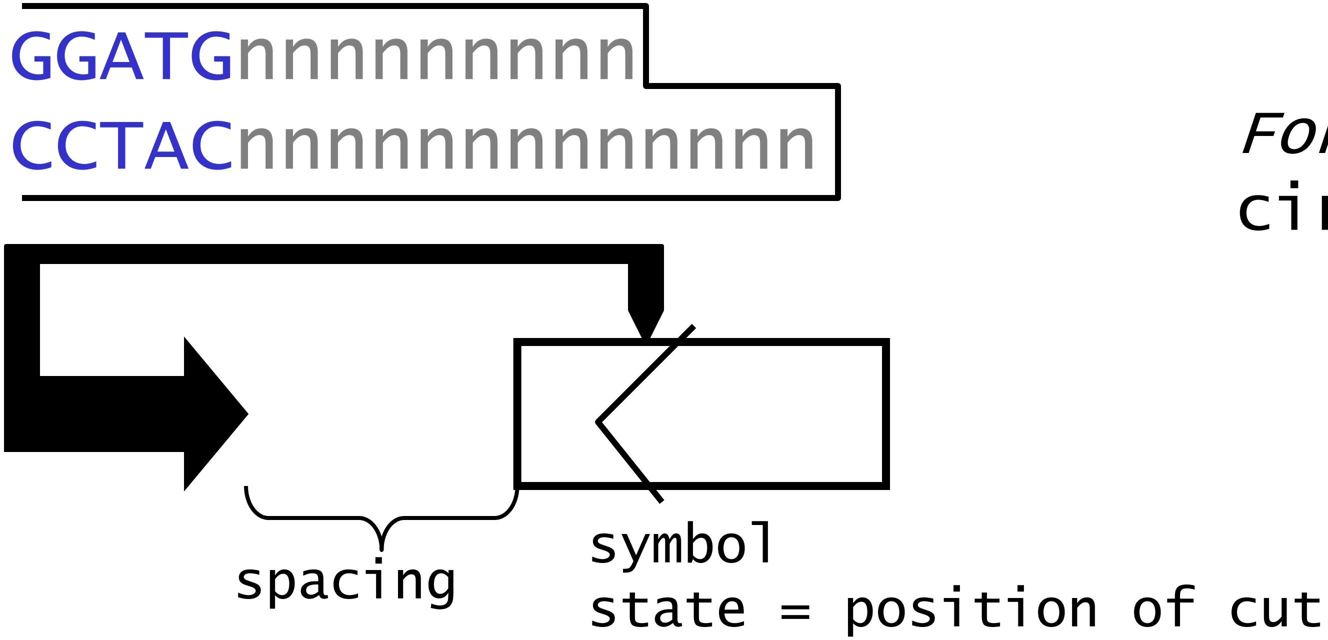


tape

1. mark a
2. move to b's
   mark b
3. move to c's
   mark c
4. if another c
5. then back to a's
   goto 1.
   else back to a's
6. check marks
   stop



'universal' Turing machine

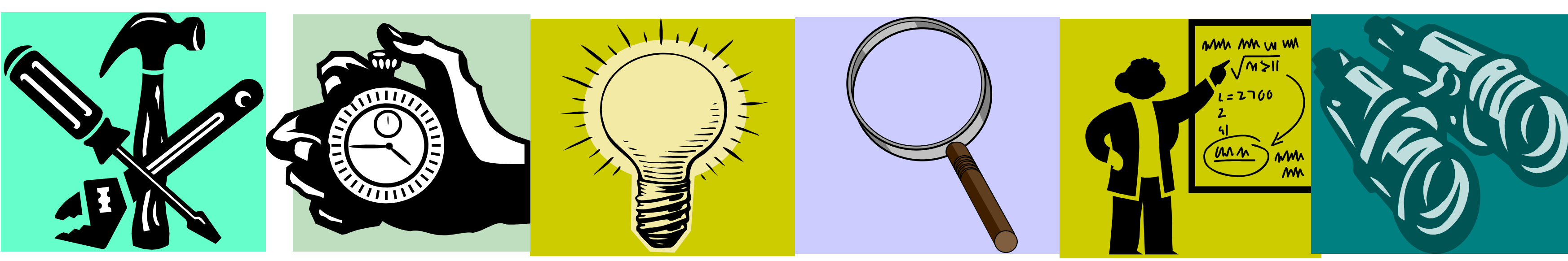


Rothemund
*Fok*I
circular DNA

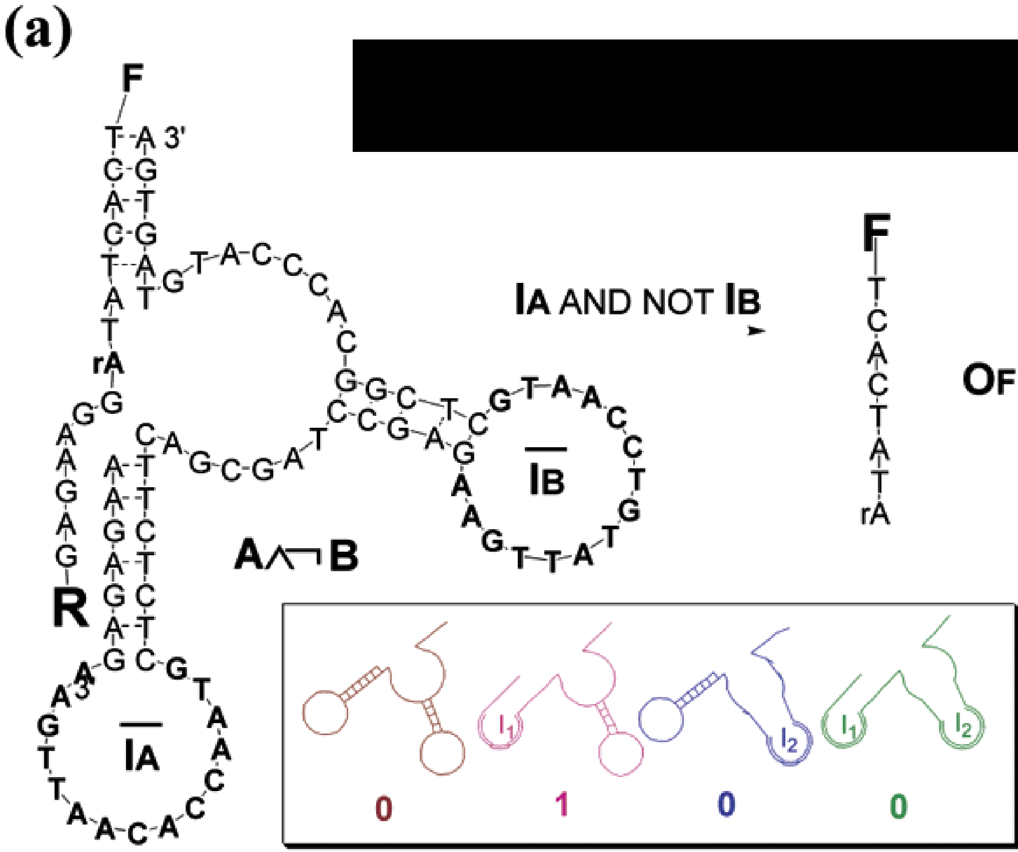
- cut states with restriction enzyme
- mix 'instructions' with 'tape'
- 'activate' instructions (cut protected end)
- ligate to form circles
- cut old symbol
- recircularize

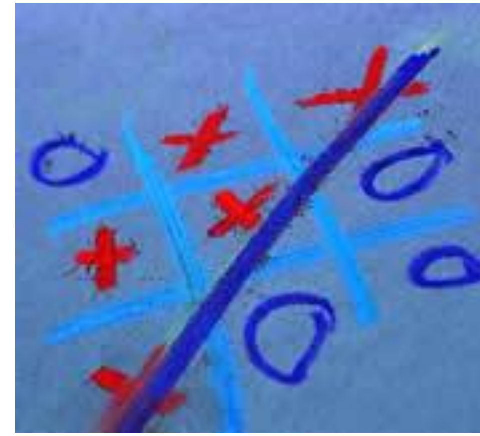
contents

- DNA … the tool chest
- Hamilton Path Problem
- Adleman’s algorithm
- comments
- theory … Turing machine
- recent work + future
- self assembly

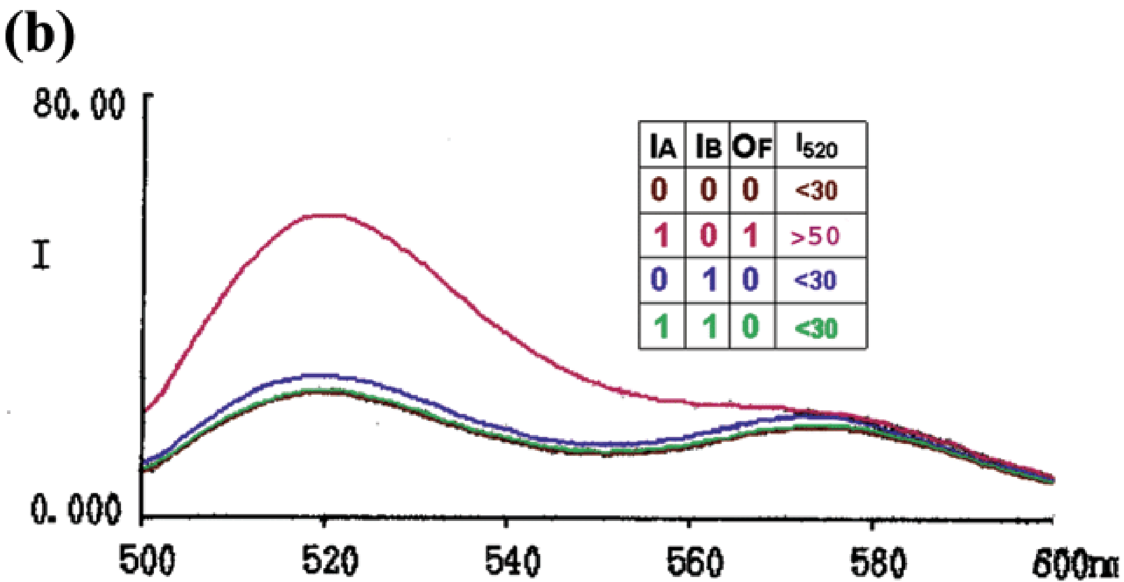


tic-tac-toe



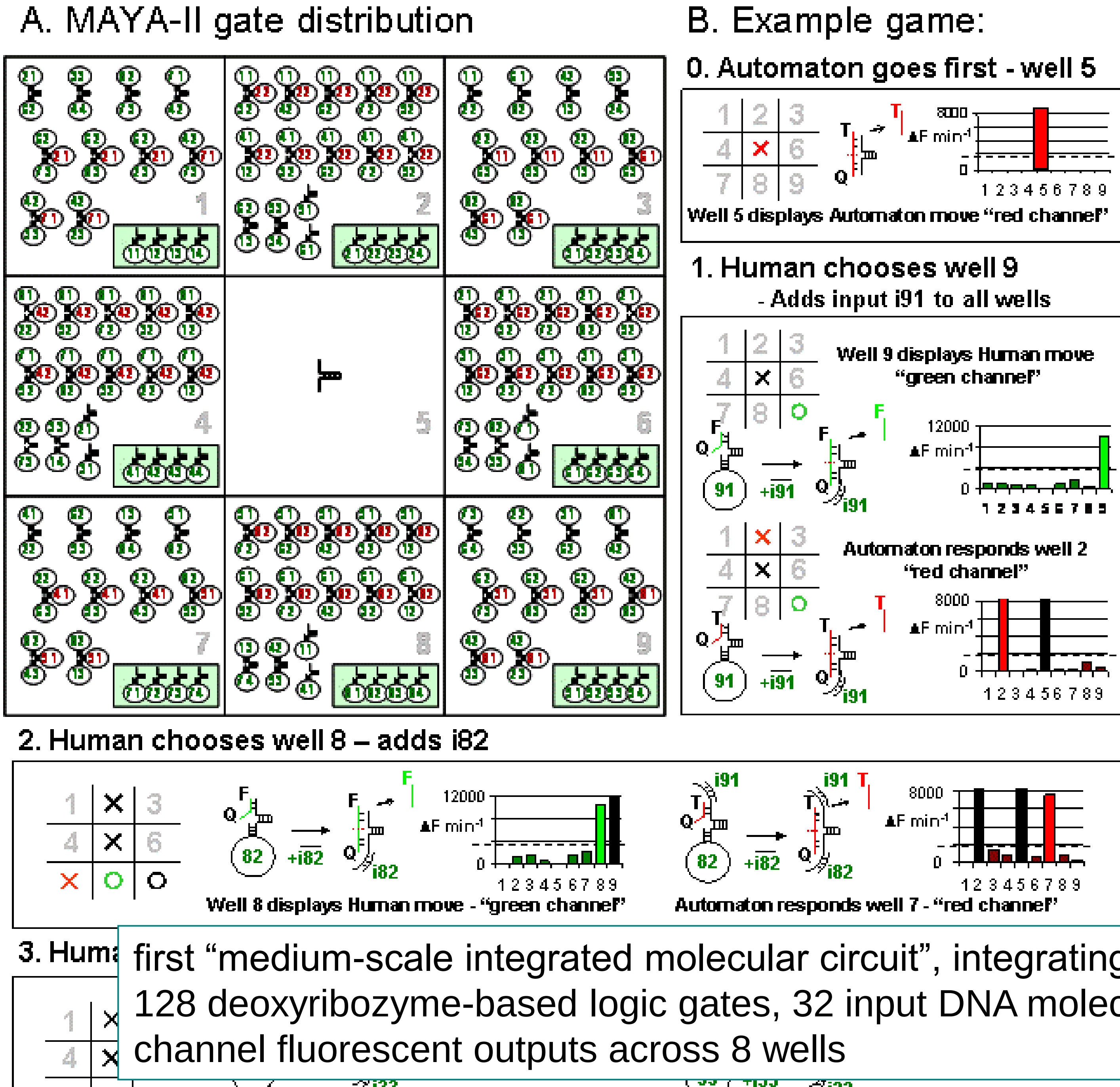


logic gates
fluorescence



IA	IB	OF	I_{520}
0	0	0	<30
1	0	1	>50
0	1	0	<30
1	1	0	<30

Stojanovic & Stefanovic, **Deoxyribozyme-Based Molecular Automaton**. *Nature Biotechn.* 2003. **Deoxyribozyme-Based Logic Gates** *J. Am. Chem. Soc*. 2002. **Medium Scale Integration of Molecular Logic Gates in an Automaton** *Nano Letters* 2006.

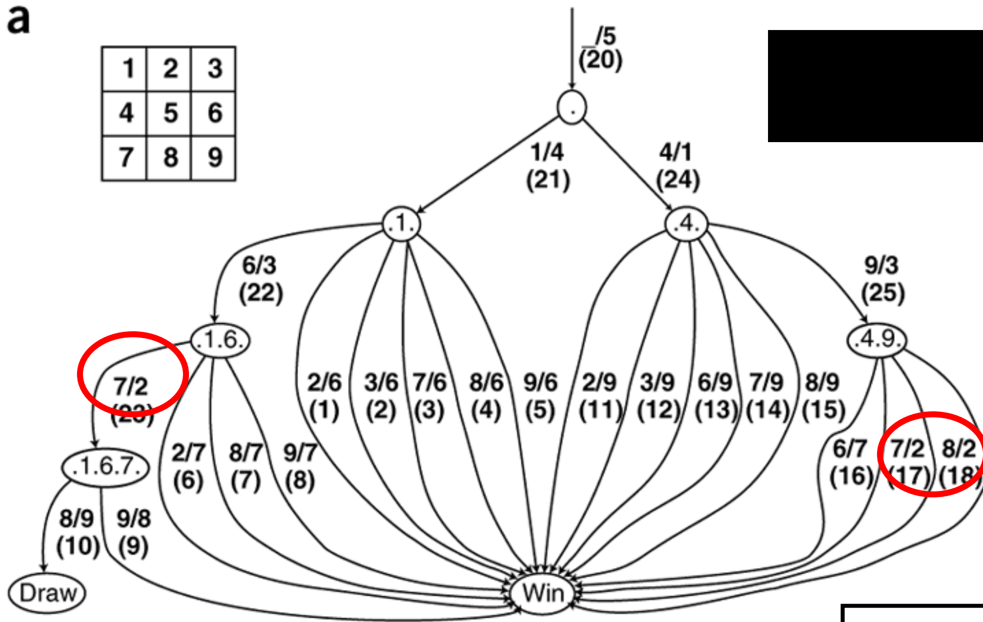


first "medium-scale integrated molecular circuit", integrating 128 deoxyribozyme-based logic gates, 32 input DNA molecules, and 8 two-channel fluorescent outputs across 8 wells

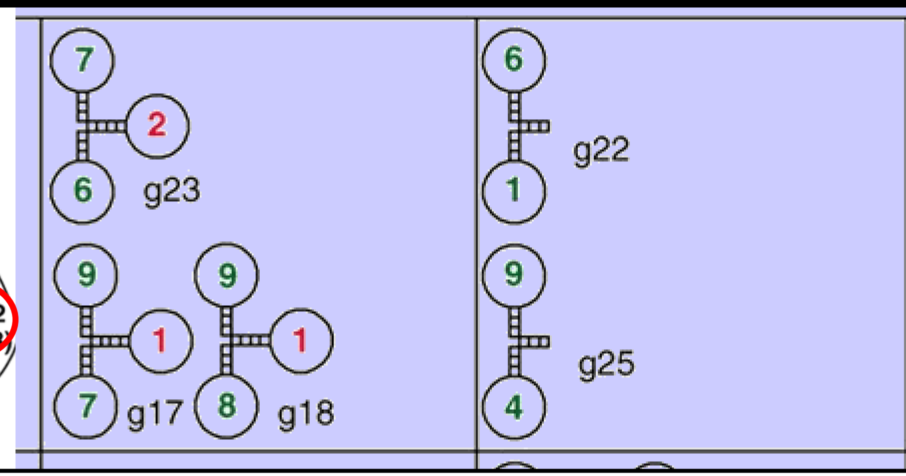
MAYA-II

digamma.cs.unm.edu/wiki/bin/view/McogPublicWeb/MolecularAutomataMAYAII

a



tic-tac-toe



$$o_2 = (i_6 \wedge i_7 \wedge \neg i_2) \vee (i_7 \wedge i_9 \wedge \neg i_1) \vee (i_8 \wedge i_9 \wedge \neg i_1)$$

b

$$o_1 = \underbrace{i_4}_{\text{edge (24)}}$$

$$o_2 = \underbrace{(i_6 \wedge i_7 \wedge \neg i_2)}_{\text{edge (23)}} \vee \underbrace{(i_7 \wedge i_9 \wedge \neg i_1)}_{\text{edge (17)}} \vee \underbrace{(i_8 \wedge i_9 \wedge \neg i_1)}_{\text{edge (18)}}$$

$$o_3 = \underbrace{(i_1 \wedge i_6)}_{\text{edge (22)}} \vee \underbrace{(i_4 \wedge i_9)}_{\text{edge (25)}}$$

$$o_4 = \underbrace{i_1}_{\text{edge (21)}}$$

$$o_5 = \underbrace{1}_{\text{edge (20)}}$$

$$o_6 = \underbrace{(i_1 \wedge i_2 \wedge \neg i_6)}_{\text{edge (1)}} \vee \underbrace{(i_1 \wedge i_3 \wedge \neg i_6)}_{\text{edge (2)}} \vee \underbrace{(i_1 \wedge i_7 \wedge \neg i_6)}_{\text{edge (3)}} \vee \underbrace{(i_1 \wedge i_8 \wedge \neg i_6)}_{\text{edge (4)}} \vee \underbrace{(i_1 \wedge i_9 \wedge \neg i_6)}_{\text{edge (5)}}$$

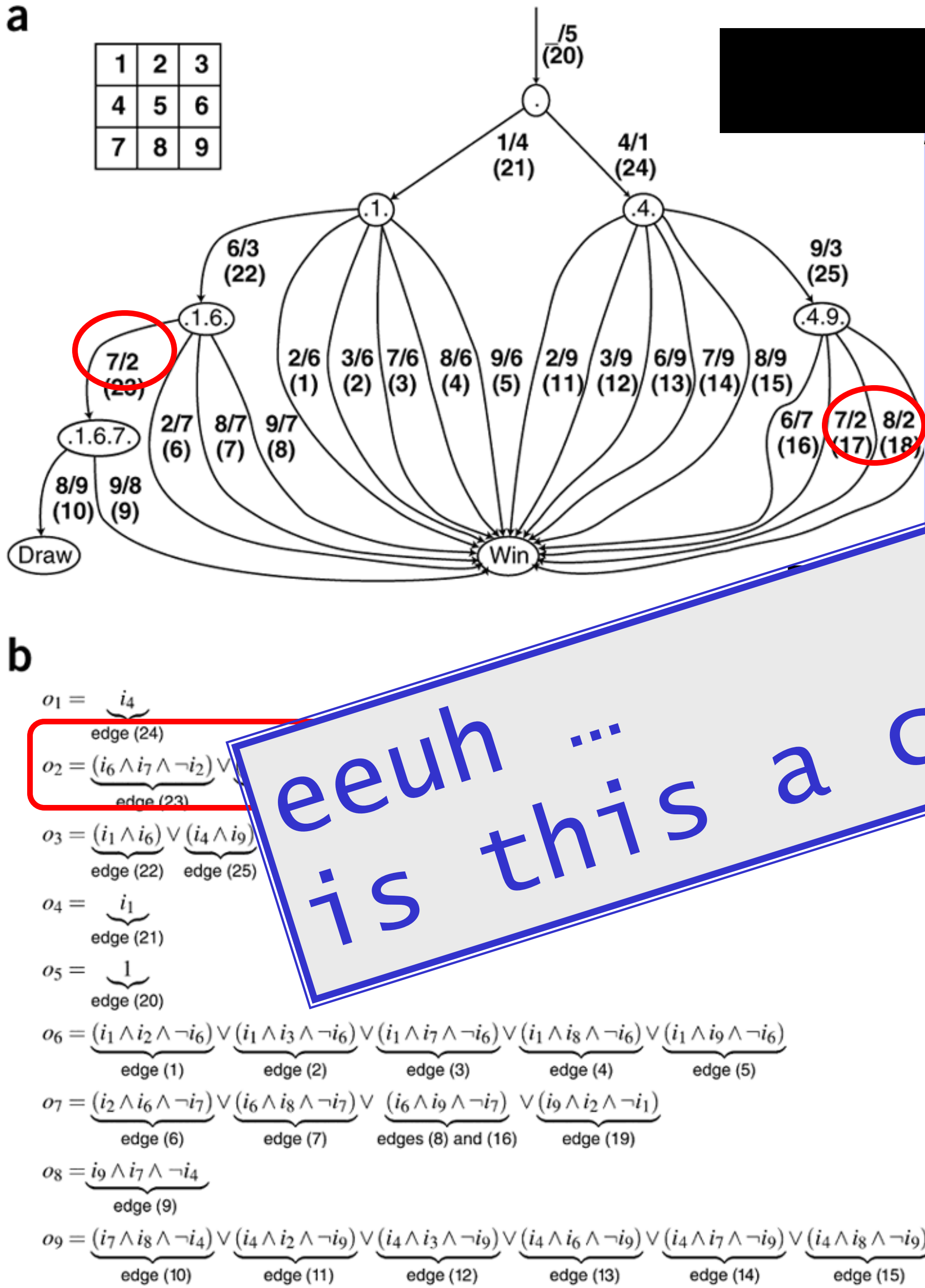
$$o_7 = \underbrace{(i_2 \wedge i_6 \wedge \neg i_7)}_{\text{edge (6)}} \vee \underbrace{(i_6 \wedge i_8 \wedge \neg i_7)}_{\text{edge (7)}} \vee \underbrace{(i_6 \wedge i_9 \wedge \neg i_7)}_{\text{edges (8) and (16)}} \vee \underbrace{(i_9 \wedge i_2 \wedge \neg i_1)}_{\text{edge (19)}}$$

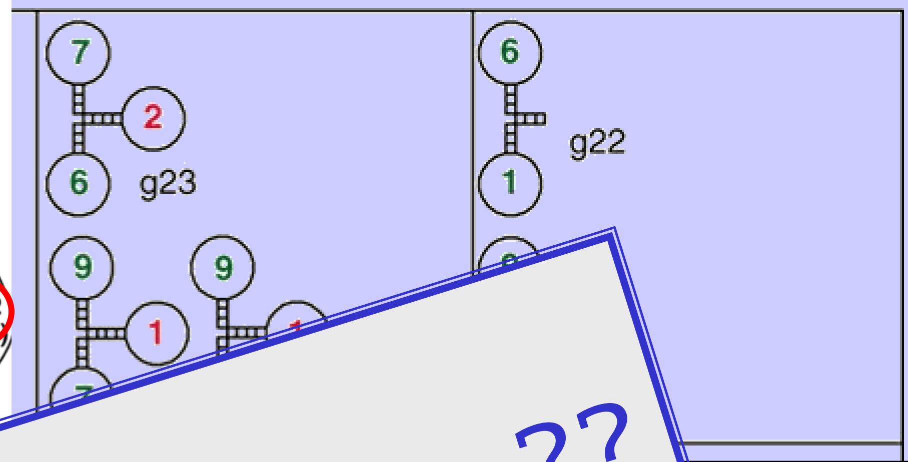
$$o_8 = \underbrace{i_9 \wedge i_7 \wedge \neg i_4}_{\text{edge (9)}}$$

$$o_9 = \underbrace{(i_7 \wedge i_8 \wedge \neg i_4)}_{\text{edge (10)}} \vee \underbrace{(i_4 \wedge i_2 \wedge \neg i_9)}_{\text{edge (11)}} \vee \underbrace{(i_4 \wedge i_3 \wedge \neg i_9)}_{\text{edge (12)}} \vee \underbrace{(i_4 \wedge i_6 \wedge \neg i_9)}_{\text{edge (13)}} \vee \underbrace{(i_4 \wedge i_7 \wedge \neg i_9)}_{\text{edge (14)}} \vee \underbrace{(i_4 \wedge i_8 \wedge \neg i_9)}_{\text{edge (15)}}$$

i_1 TCT GCG TCT ATA AAT
i_2 ATC GTA TGT TGT TCA
i_3 GTA TAG TCT GTT TGT
i_4 G TAA GTG CTC AAA TGT C
i_5 G TCT AAT TCT CAC GGT C

tic-tac-toe





$\ldots_1) \vee (i_8 \wedge i_9 \wedge \neg i_1)$

eeuh …
is this a computer ??

i_1 TCT GCG TCT ATA AAT
i_2 ATC GTA TGT TGT TCA
i_3 GTA TAG TCT GTT TGT
i_4 G TAA GTG CTC AAA TGT C
i_5 G TCT AAT TCT CAC GGT C

Tom’s links

Future directions in computing: DNA Computing, *BBC News*, 13 Nov 2007
http://news.bbc.co.uk/2/hi/technology/7085154.stm

"This soup of DNA and enzymes implements a well know mathematical model of computation known as finite automaton," he explained.
"This finite automaton knows how to do very simple computation such as recognising whether a list of zeros and ones has an even number of ones."

In the case of his 2004 computer this method of computation was used to analyze ratios of specific molecules related to prostate cancer and a specific type of lung cancer.

The "computer" consisted of a chain of three segments of DNA and an enzyme which could cut the strands.

Tom’s links

DNA computer 'ansers questions', *BBC News*, 05-Aug-2009
http://news.bbc.co.uk/2/hi/technology/8184033.stm

... they tried the system with simple "if… then…" propositions. One of these went as follows: "All men are mortal. Socrates is a man. Therefore, Socrates is mortal."

The answer was encoded in a flash of green light. Some of the DNA strands were equipped with a naturally glowing fluorescent molecule bound to a second molecule which keeps the light covered.

The system can take in facts and rules as a computer file of simple text. The robotic "compiler" can then turn those facts and rules into the DNA starting products of a logical query.

In other words, computers that go to work inside a cell.

Tom's links

DNA computer 'ansers questions', *BBC News*, 05-Aug-2009
http://news.bbc.co.uk/2/hi/technology/8184033.stm

Tom's links

DNA logic gates herald injectable computers, *New Scientist*, 02 June 2010
http://www.newscientist.com/article/dn18989-dna-logic-gates-herald-injectable-computers.html

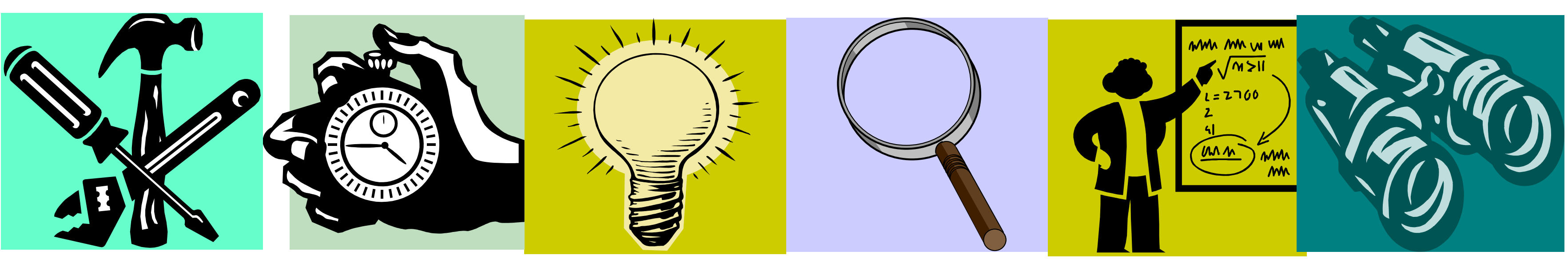
"The biocomputer would sense biomarkers and immediately react by releasing counter-agents for the disease," says Itamar Willner, who led the work.

The new logic gates are formed from short strands of DNA and their complementary strands, …. Two strands act as the input: each represents a 1 when present or a 0 when absent. … Take the "exclusive OR" or XOR logic gate. It produces an output when either of the two inputs is present but not when both are present or both are absent.

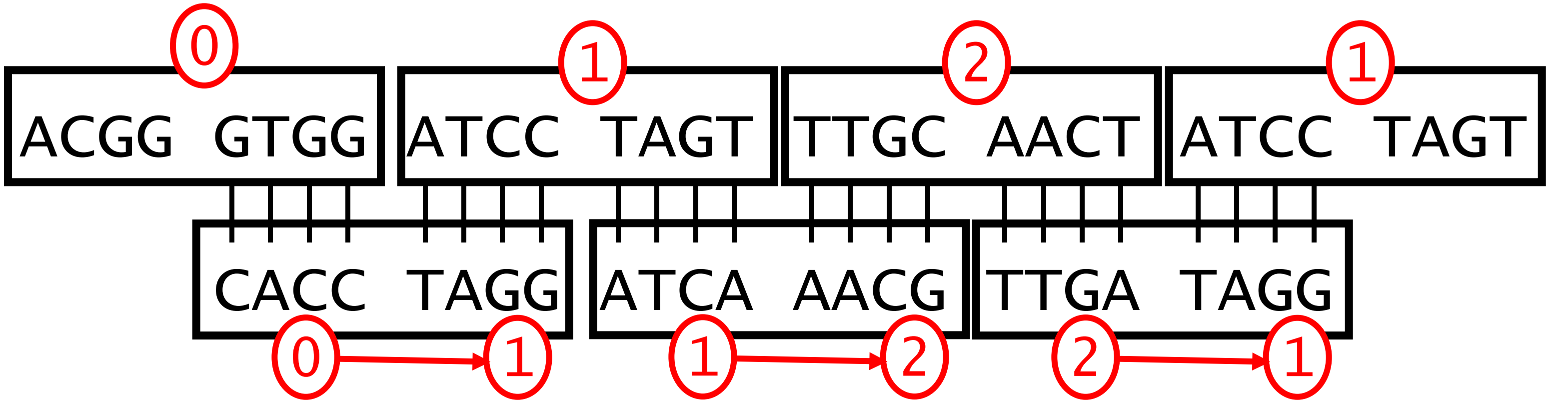
Willner and his team added molecules to both the complementary strands that caused them to fluoresce when each was present in isolation, representing a logical 1 as the output. But when both were present, the complementary strands combined and quenched the fluorescence, representing a 0 output.

contents

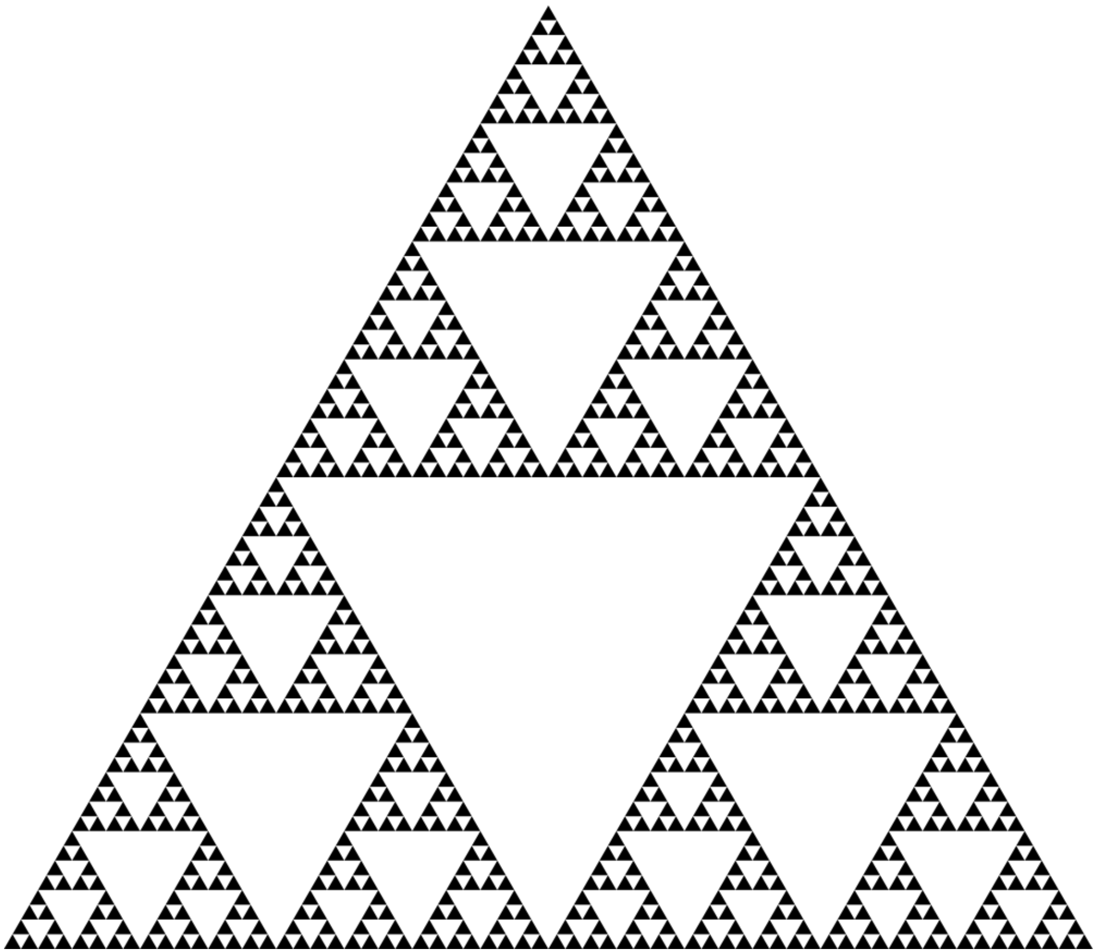
- DNA … the tool chest
- Hamilton Path Problem
- Adleman's algorithm
- comments
- theory … Turing machine
- recent work + future
- self assembly



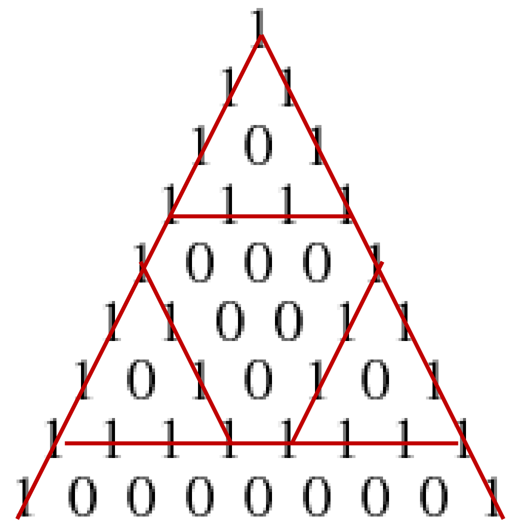
self assembly



Sierpinski triangle



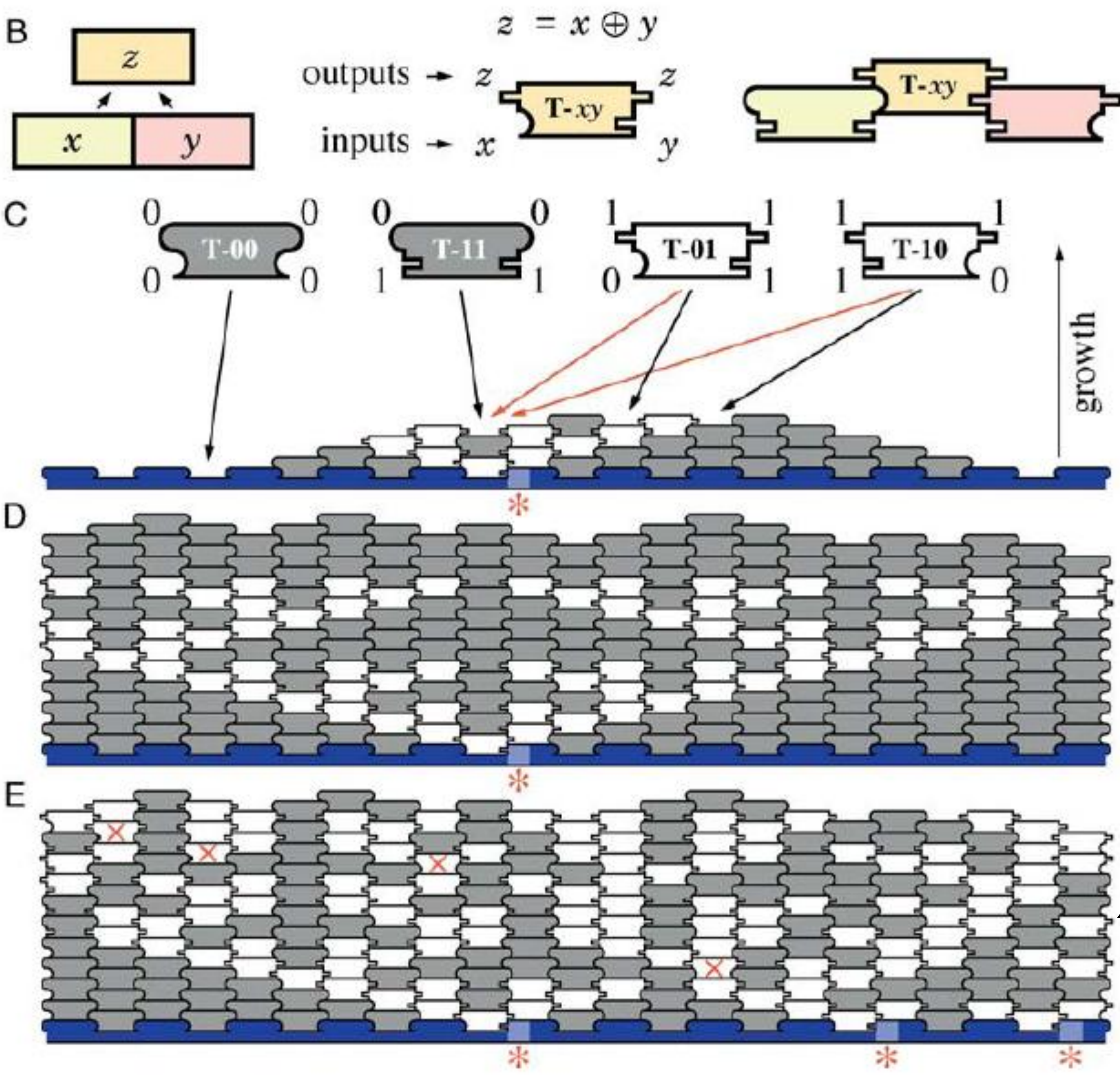
‘fractal’

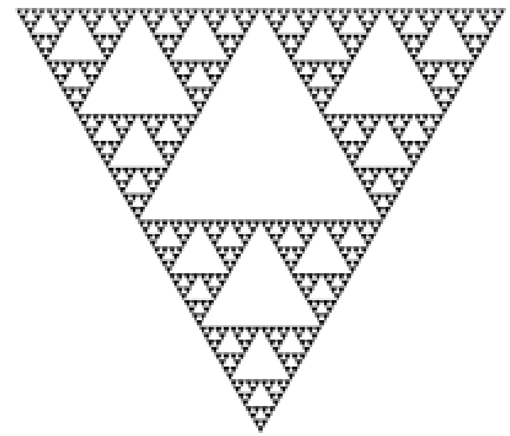


Sierpinski triangle

$\oplus$ XOR
even / odd

self assembly: Sierpinski

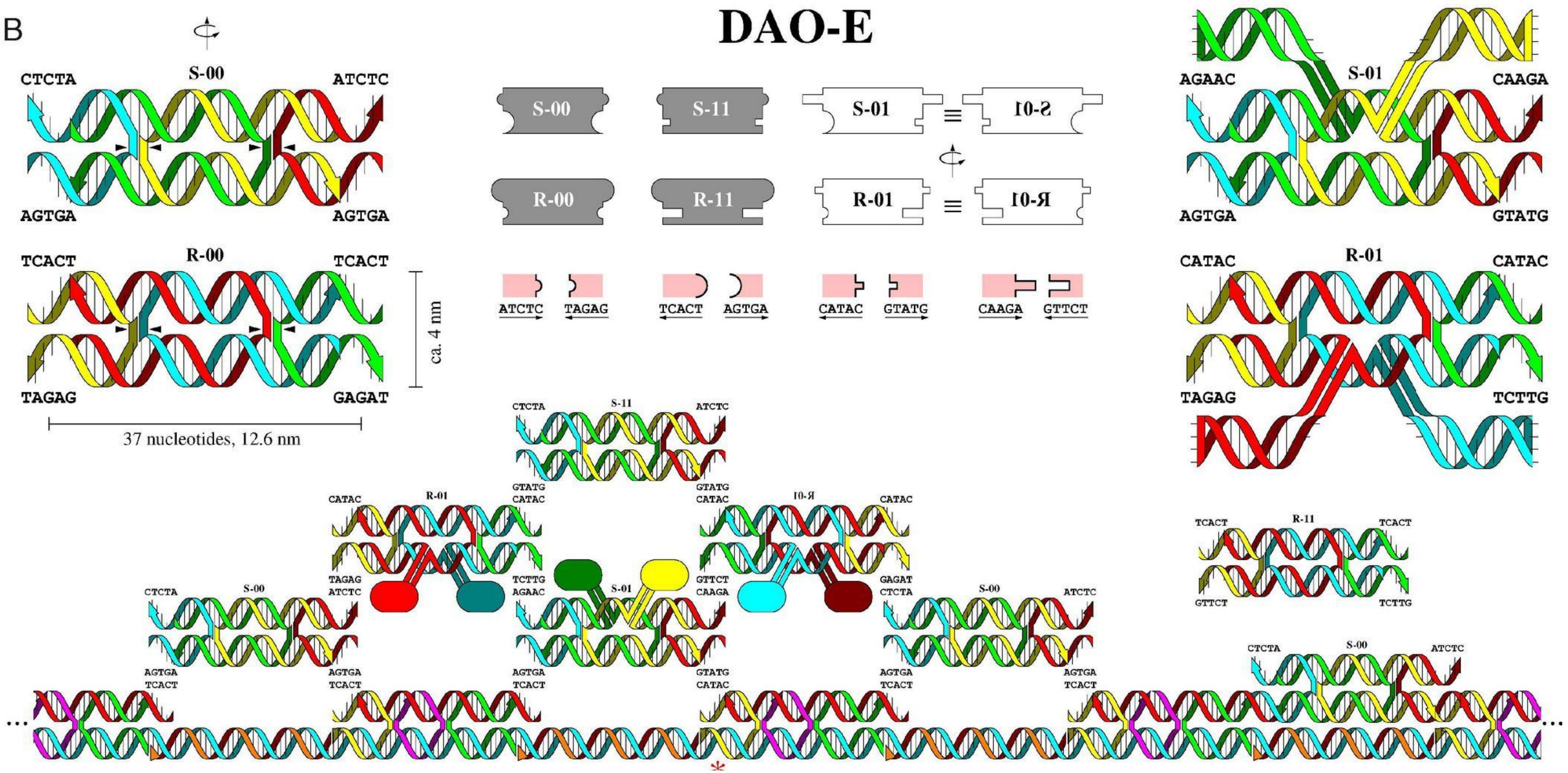




Sierpinski

Algorithmic Self-Assembly of DNA Sierpinski Triangles, Rothemund, Papadakis, Winfree; PLoS Biology (2004)

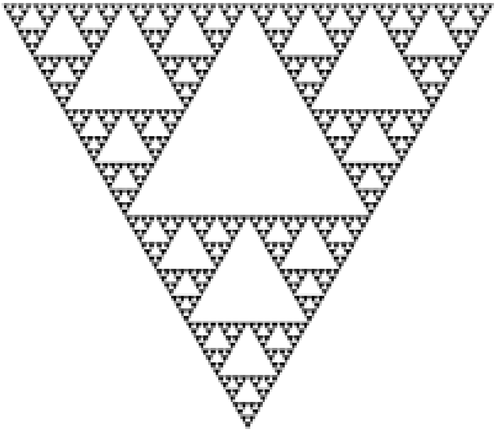
self assembly



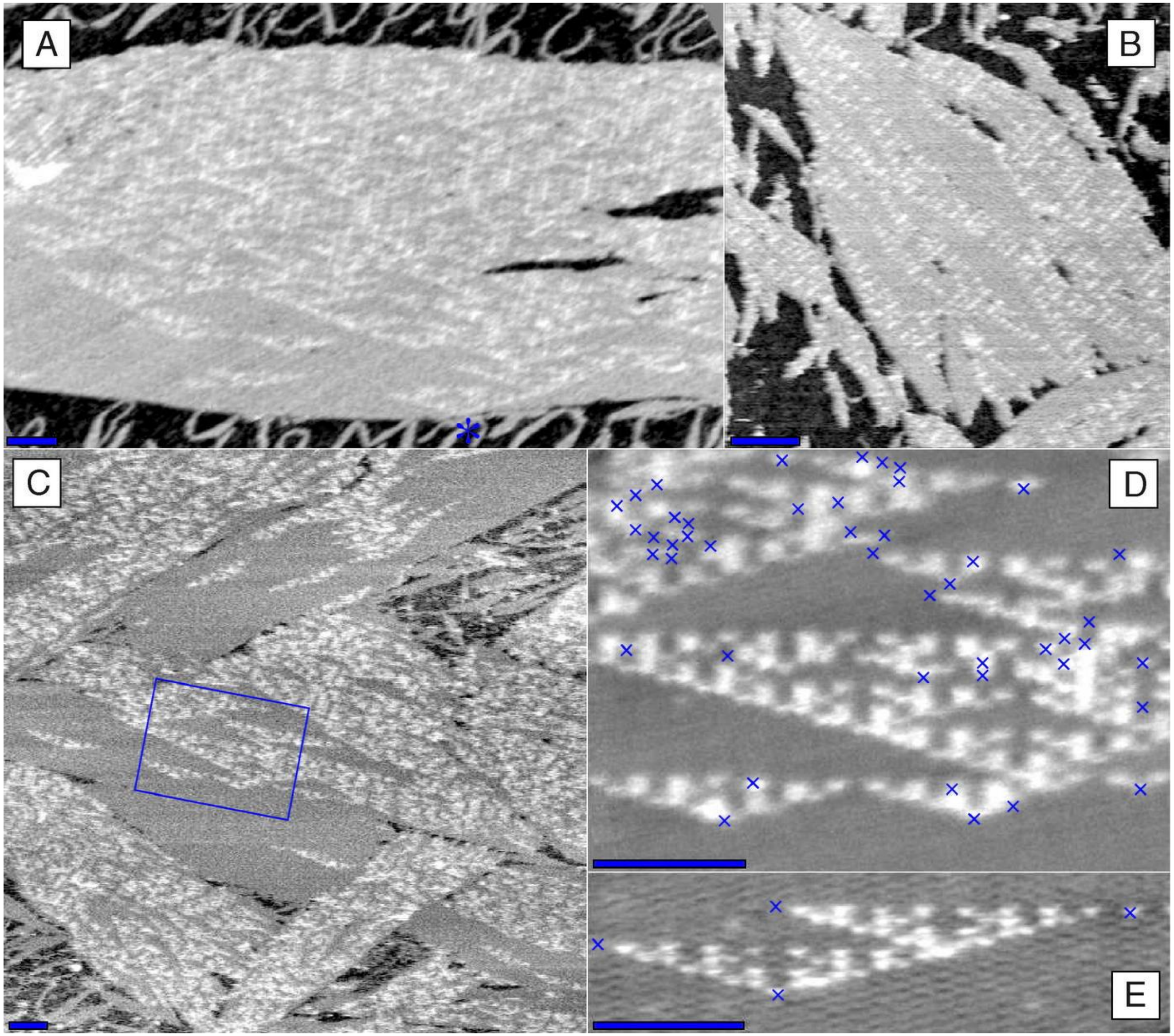
http://dx.doi.org/10.1371/journal.pbio.0020424

Algorithmic Self-Assembly of DNA Sierpinski Triangles
Rothemund, Papadakis, Winfree; PLoS Biology (2004)

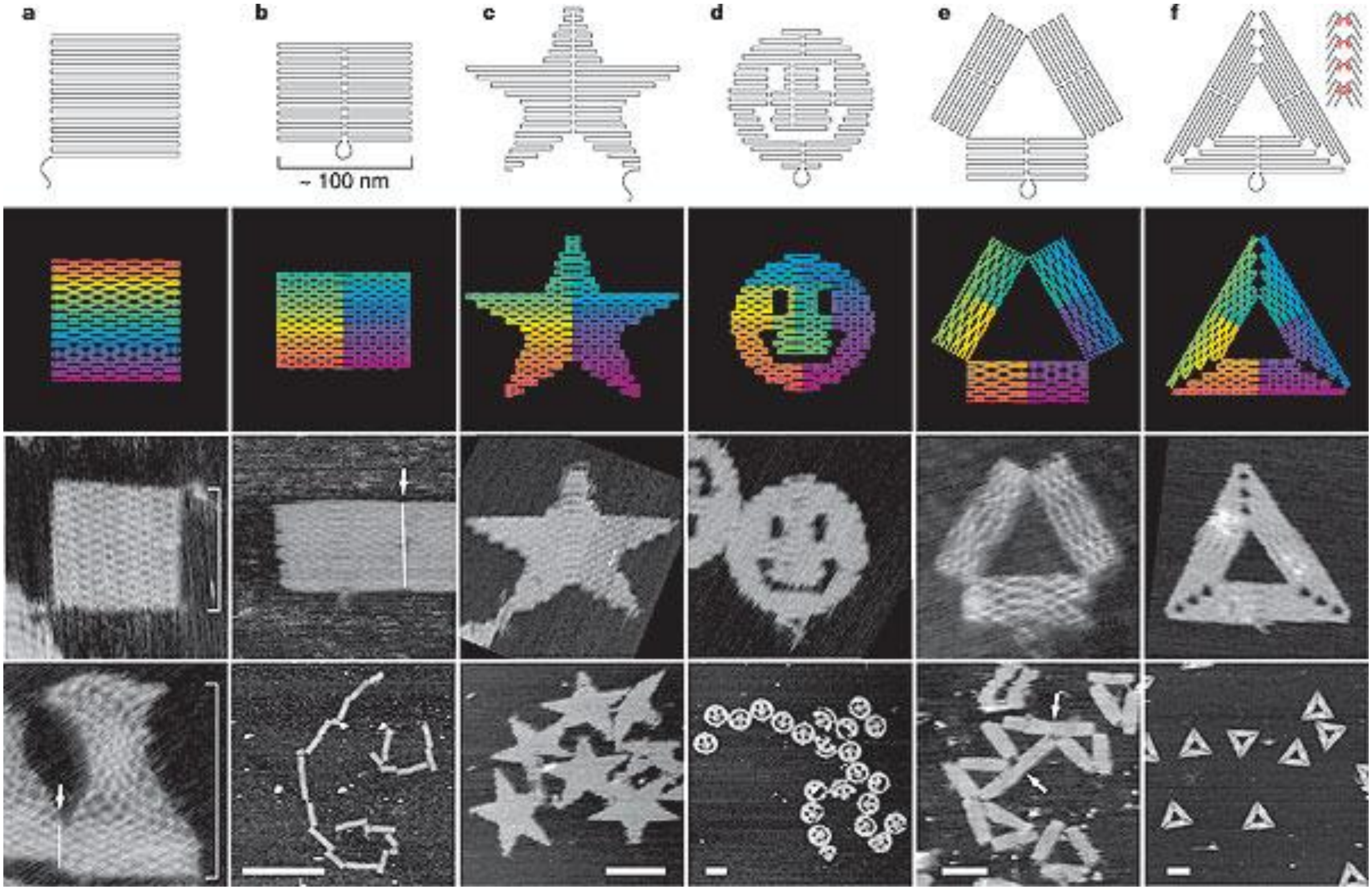
self assembly



Sierpinski

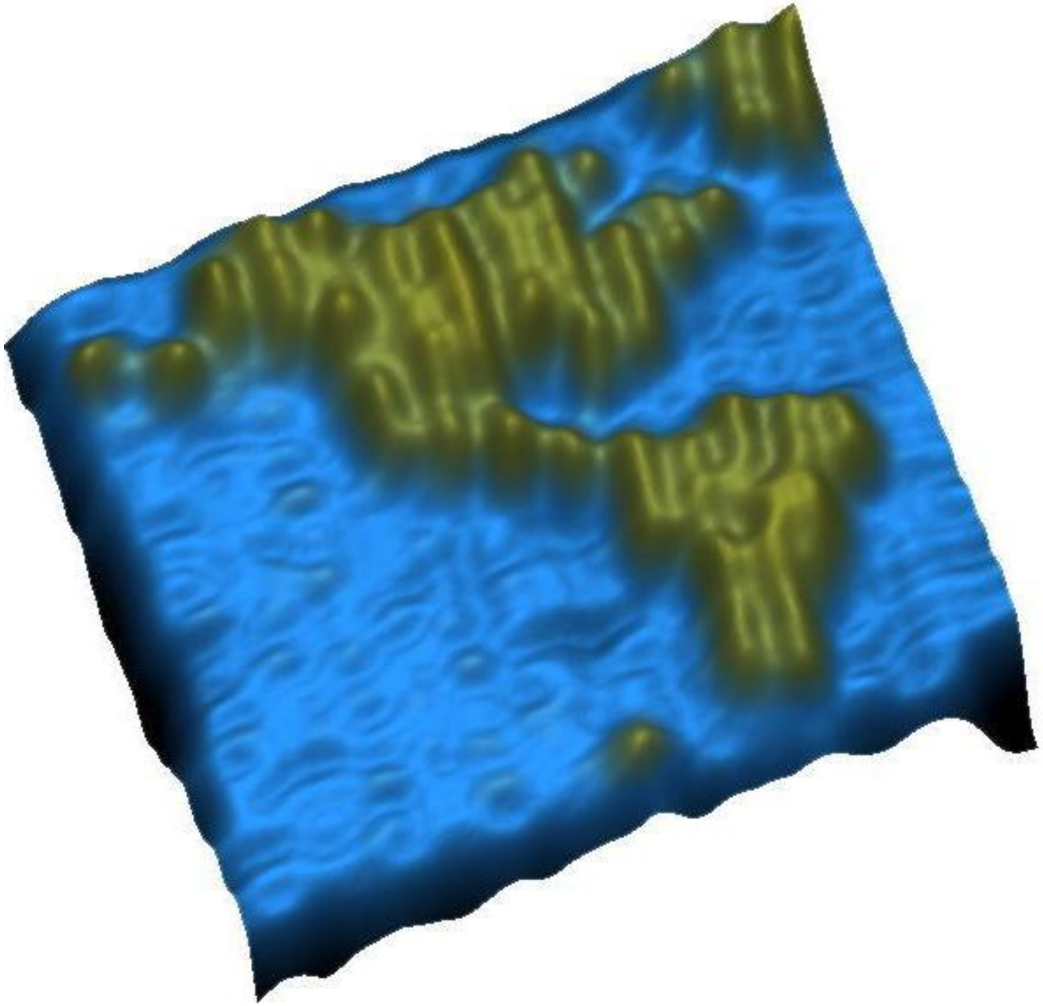


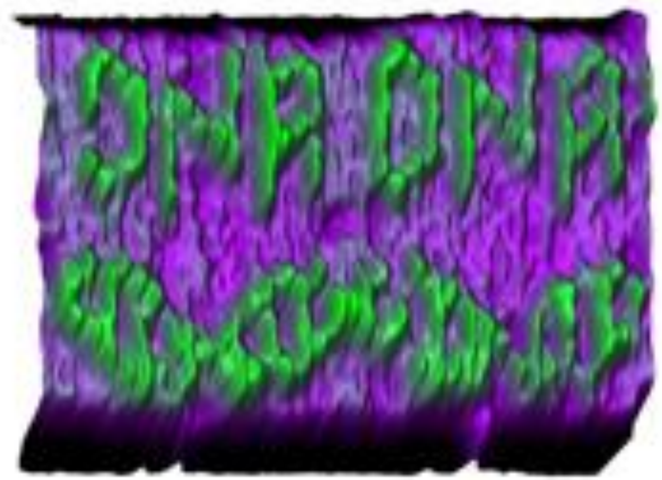
self assembly: DNA origami



Folding DNA to create nanoscale shapes and patterns
Paul W. K. Rothemund, Nature 440, 297-302 (16 March 2006)

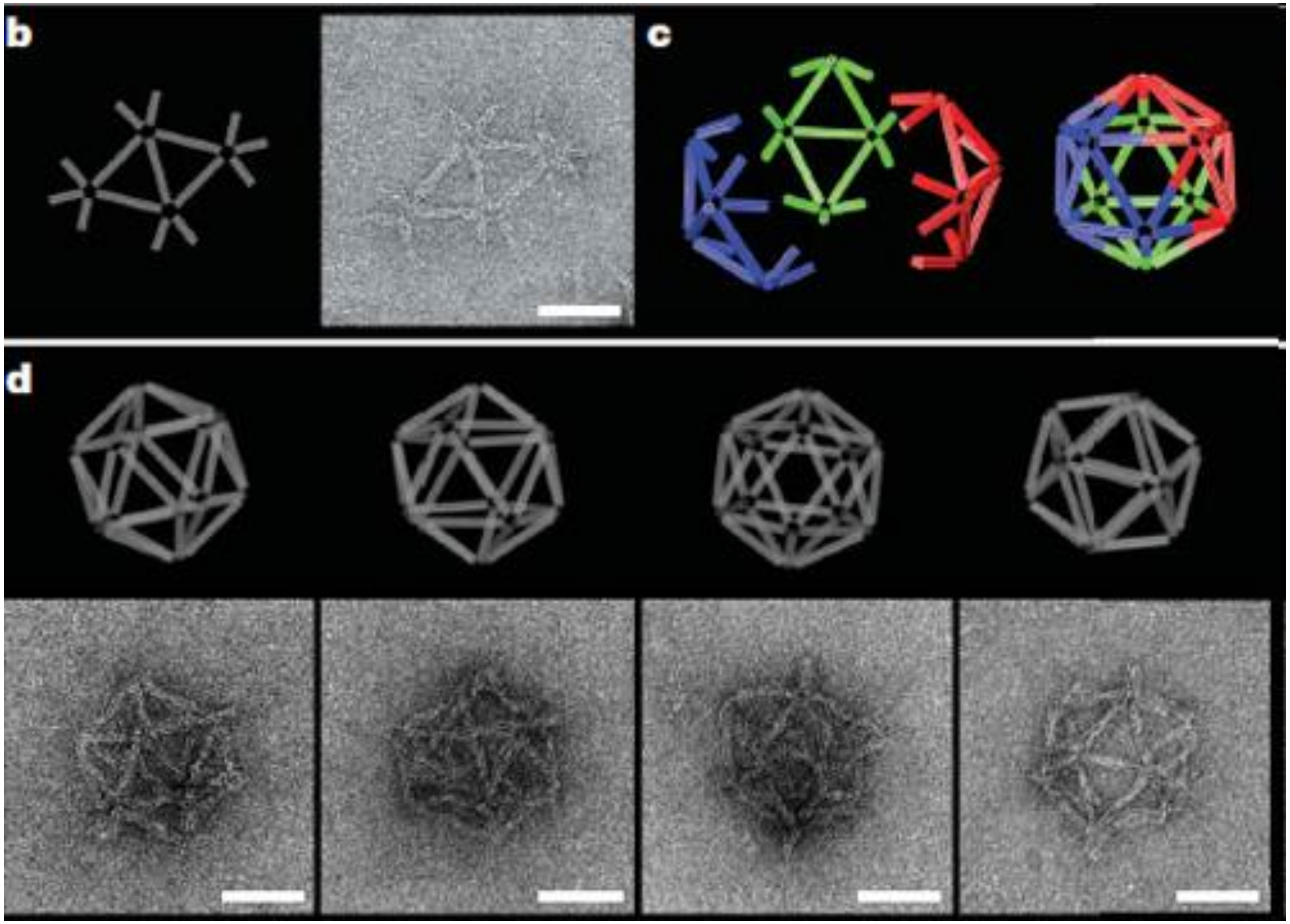
Self Assembly: DNA origami





Paul W. K. Rothemund, http://www.dna.caltech.edu/~pwkr/

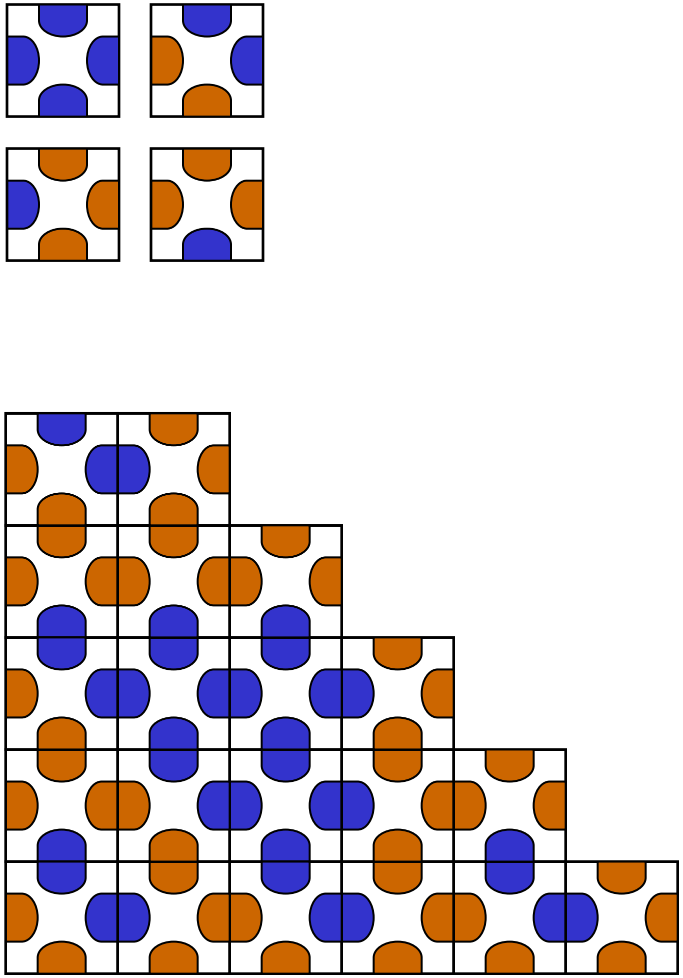
3D DNA origami



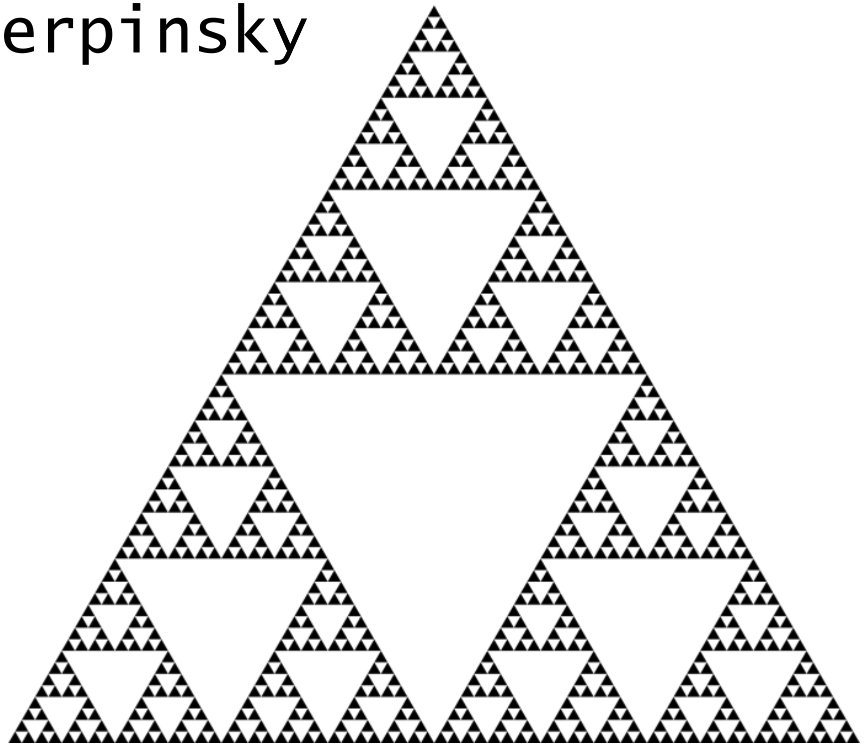
Self-assembly of DNA into nanoscale three-dimensional shapes
S.M. Douglas, H. Dietz, T. Liedl, B. Hogberg, F. Graf, W.M. Shih,
Nature 459, 414-418 (21 May 2009)

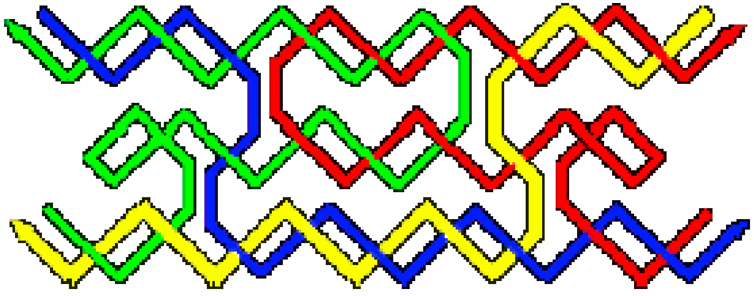
self assembly (theory)

Wang tiles



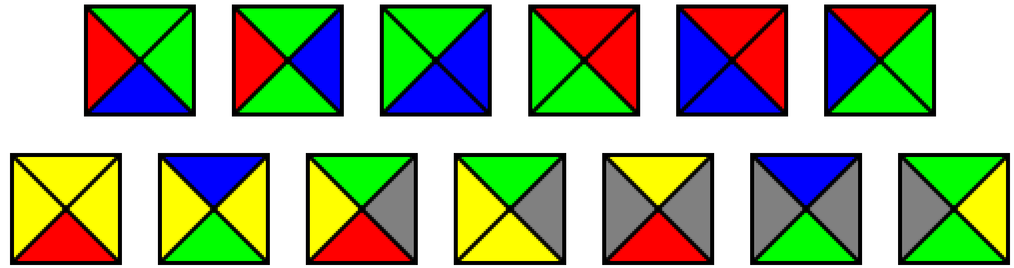
Sierpinsky

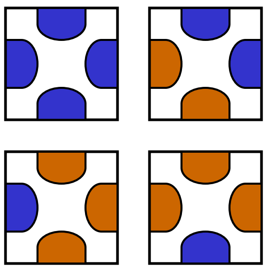


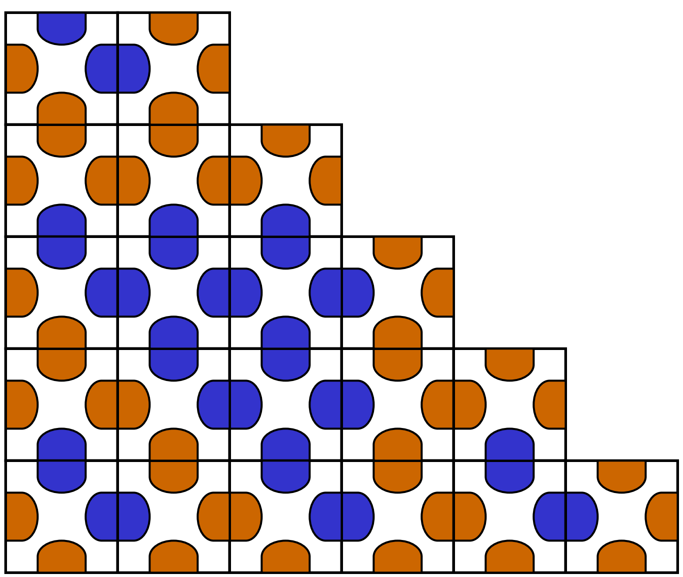


self assembly

Wang tiles



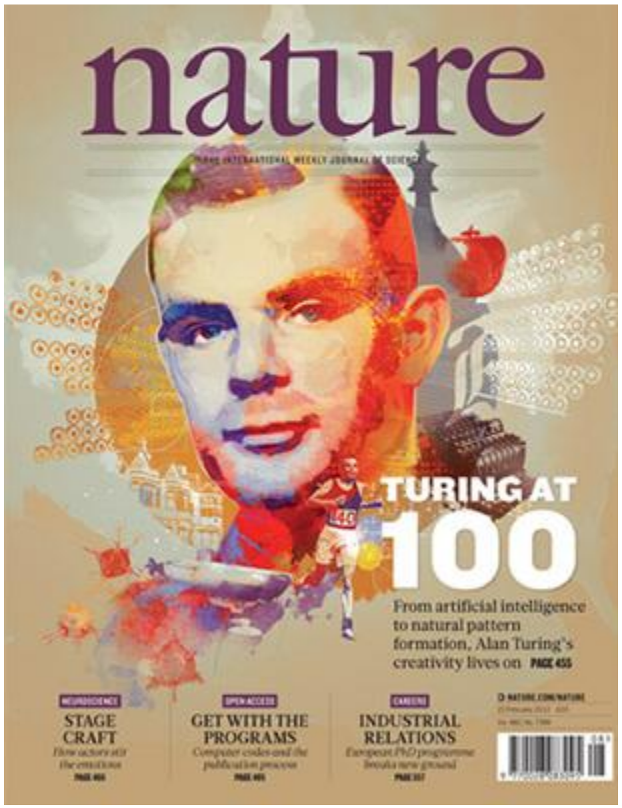


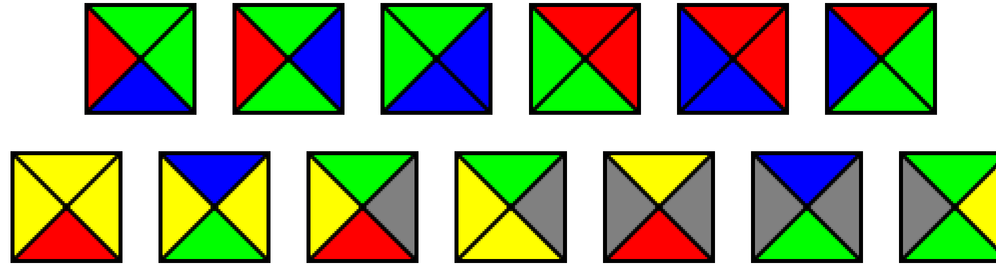


can we tile
the plane?

undecidable

self assembly



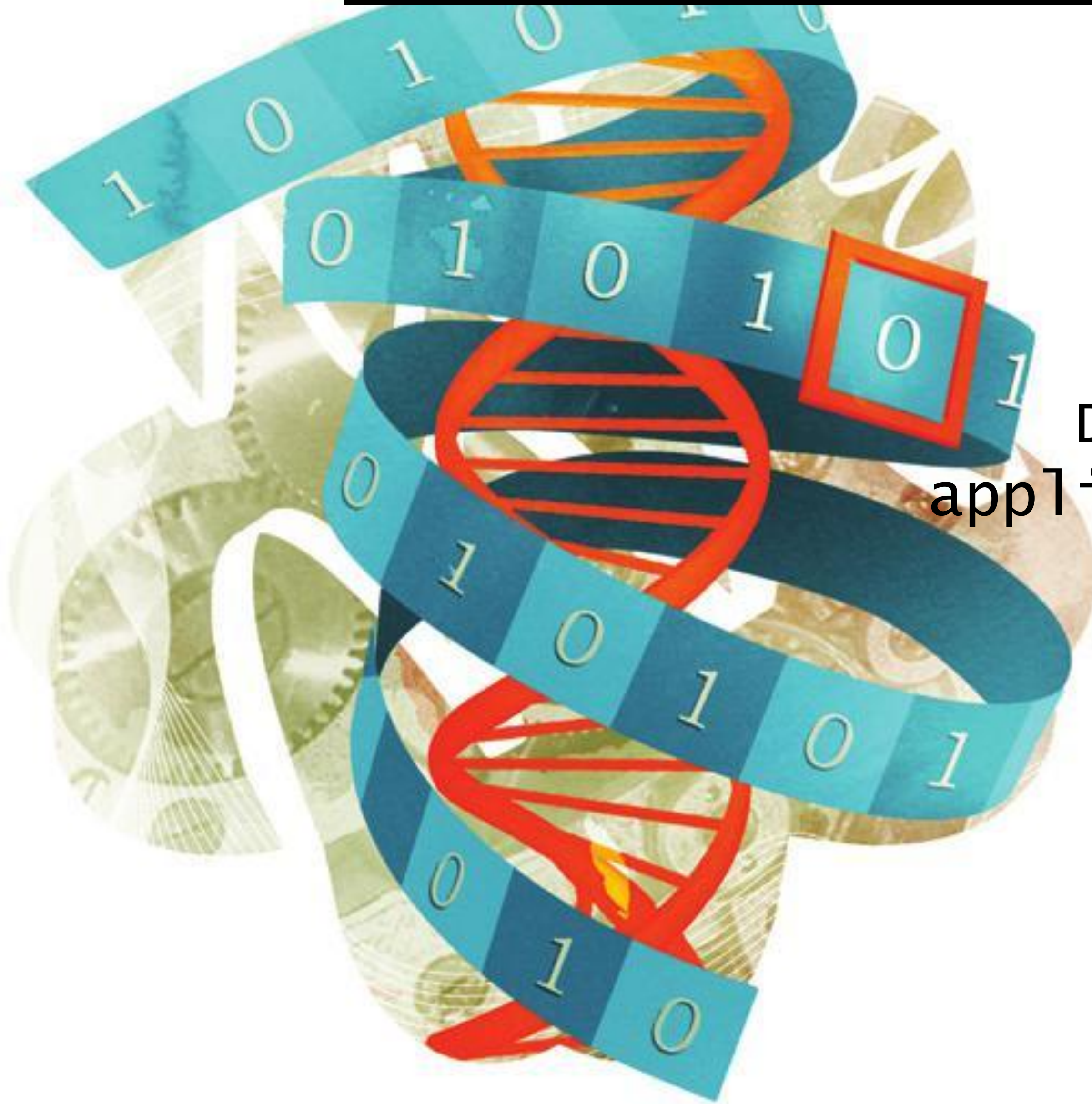


can we tile the plane? — undecidable

rectangle — NP-complete

strip — PSPACE-compl

conclusion



take home message

DNA can be used for applications it was not “intended” for

computing
a very interesting proof of concept

find niche