It's All About Programming, Stupid!

LaTiCE 2016, IIT Bombay
2 April, 2016

Madhavan Mukund
Chennai Mathematical Institute
http://www.cmi.ac.in/~madhavan

Programming and CS

- There is more to computer science than programming …
- … but if you can't program effectively, you won't make it as a computer scientist

Computational Thinking

- "… whether learning coding is enough to build computational thinking and knowledge …"
- "… computer science as an academic discipline is larger than coding—and in the haste to teach programming skills, the problem-solving and creative thinking that undergirds technical innovations is lost."

The Atlantic, 29 February, 2016

Computational Thinking

"… the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent."

Computational Thinking …

- "Effectively carried out by … an agent"
  - Spell out the rules of the game
  - What actions can the agent perform?
  - What resources are available?
- Cannot teach this in the abstract

Teaching programming

- C, C++, Java are syntax heavy
  - Declarations, return types, keywords,…
  - Problem-solving takes a back seat
- Alice and Scratch for younger students
- Move up to Python et al?

Teaching programming …

- Is this exciting?
    - Convert fahrenheit to centigrade
    - Add up the digits in a number
    - Compute tax based on income slabs
- Wow??

Simple and interesting?

- Simplecpp—Abhiram Ranade
- Primitives for drawing: “Turtle” graphics
- repeat( ), other wrappers to reduce clutter
- Draw on familiar, non-trivial examples
  - Algebra, physics, geometry, art …

The Simplecpp manifesto

“It may seem that the agenda of getting to many applications in science and technology might conflict with the (primary?) goal of conveying the basic computer programming concepts. However, by choosing the examples carefully, we can in fact motivate even the programming concepts better.”

Starting younger

- Non-trivial algebra, physics is taught only in high school
- What can we introduce in Class 8?
  - Graphs
  - Inductive definitions
  - Recursion

Informatics Olympiad

- Algorithms and programming contest for high school students
- C, C++, Pascal, Java
- Dynamic memory allocation not required
    - All problems can be solved using fixed-size arrays

Informatics Olympiad …

- Within this limited framework …
  - Sorting, searching, graph algorithms, dynamic programming
  - Data structures such as heaps, range trees
- Students typically exploit more features
  - STL library, vectors, maps, …

Indian Computing Olympiad

- Two rounds of selection exams followed by a two week training camp—since 2002
- Camp used to start with sorting and work through graphs, dynamic programming, data structures
  - Taught through "case studies"
- Today, camp students know it all already
  - Focus on problem solving

ICO …

- Students are typically self-taught
- Computer science in school is “boring”
  - Memorize lots of facts, like biology!
  - Practicals are from a fixed set of programs
  - Teachers are not adequately trained

ICO …

- These students go on to join CS programmes in the best institutions world-wide
  - Incentives in CMI, IIIT Hyderabad, IIIT Delhi
- Overwhelming number remain in CS after college

Teaching algorithms

- Typical progression
  - Introduction to programming
  - Data structures
  - Design and analysis of algorithms
- No meaningful programming for algorithms course

Appreciating algorithms

- Need to work with life-size examples
  - To sort 100 values, any algorithm works
  - To sort 1 billion = 10^9 SIM cards in India
    - Bubble sort takes 10^9 seconds = 300 years
    - Merge sort takes under 10 minutes

Understanding O( )

Input	log n	n	n log n	n^2	n^3	2^n	n!
10	3.3	10	33	100	1000	1000	10^6
100	6.6	100	66	10^4	10^6	10^{30}	10^{157}
1000	10	1000	10^4	10^6	10^9		
10^4	13	10^4	10^5	10^8	10^{12}		
10^5	17	10^5	10^6	10^{10}			
10^6	20	10^6	10^7				
10^7	23	10^7	10^8				
10^8	27	10^8	10^9				
10^9	30	10^9	10^{10}				
10^{10}	33	10^{10}					

Understanding O( )

Input	log n	n	n log n	n^2	n^3	2^n	n!
10	3.3	10	33	100	1000	1000	10^6
100	6.6	100	66	10^4	10^6	10^{30}	10^{157}
1000	10	1000	10^4	10^6	10^9		
10^4	13	10^4	10^5	10^8	10^{12}		
10^5	17	10^5	10^6	10^{10}			
10^6	20	10^6	10^7				
10^7	23	10^7	10^8				
10^8	27	10^8	10^9				
10^9	30	10^9	10^{10}				
10^{10}	33	10^{10}					

1 second = 10^8 operations

Using data structures

- Heaps : delete-minimum, insert
- Algorithms : Dijkstra's single-source shortest path can be more efficient using heaps
- Connect the dots: Dijkstra's algorithm requires updating heap values in place
    - Swept under the carpet

At CMI

- Two programming courses plus algorithms form a continuum
- Haskell → Python → Algorithms
  - Functional programming without religion!
  - Sorting, graph algorithms, dynamic programming, heaps, balanced binary trees in first two courses, with coding

Beyond algorithms

- Automata theory is all about programming
- Again, syntax obfuscates the true picture
  - "A DFA is a five-tuple $(Q,\Sigma,\partial,q_0,F)$ …"

Automata theory

- Given L regular, {w | ww in L} is regular
  - Simulate the automaton for L
  - Use "pebbles" as markers
- Operational proofs, not existential arguments

Automata theory …

- Need to play with the model to understand it
- True power of nondeterminism is tricky to grasp
- Abstraction can only come with operational understanding
  - What the model can do, and cannot do

Other subjects

- Databases
  - Large-scale, interesting queries in SQL
  - All one-change routes on Indian Railways
- Concurrency
  - Locks, semaphores, monitors through examples—“puzzles”

On a different note

- Even the most theoretical of computer scientists are more practical than you think!
  - Can program, but choose not to
  - But, when push comes to shove …

TikZ—pictures in LaTeX

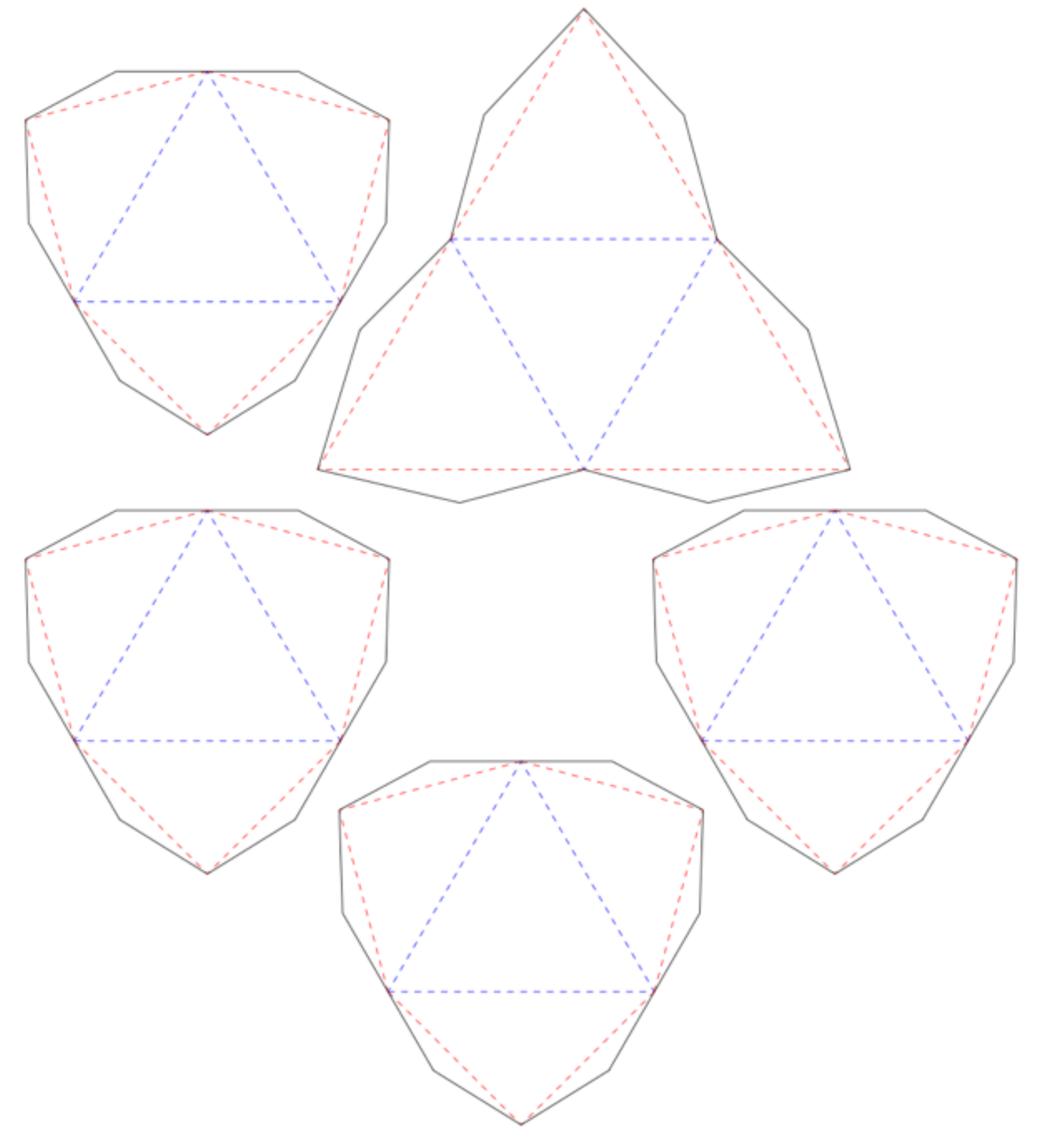
```
%% Locate  vertices of the flaps:
%% Rotate each edge  through 15% and scale by 1/2 cos(15)
%% Clockwise gives e, counter-clockwise gives f
%% Draw the external lines

\foreach \i in {1,2,3}
 \coordinate  (e\i\k) at ($(a\i\k) ! 0.5*cos(15) ! -15:(d\i\k)
$);

\foreach \i/\j in {1/2,2/3,3/1}
 \coordinate  (f\i\k) at ($(a\j\k) ! 0.5*cos(15) ! 15:(d\i\k)$);

\foreach \i/\j in {1/2,2/3,3/1}
{\draw (a\i\k) -- (e\i\k) -- (d\i\k);
\draw (a\j\k) -- (f\i\k) -- (d\i\k);
```

Five tetrahedra that can be assembled into a cube



Algebraic complexity theory

```
shaplists=LMgetshaplist(finmylist);
[ig szfinmylist]=size(finmylist);
for kk=1:1:szfinmylist
     Hwtlist{kk}={};
[ig, szshapes]=size(shaplists);
[ig,szlistm]=size(listmwts);
[ig,szlistn]=size(listnwts);
outercount=1;
if (nargin==1)
  for kk=1:1:szlistm
     wtm=listmwts{kk};
innercount=0;
for jj=1:1:szlistn
     wtn=listnwts{jj};
obj=LMgl3highestweights(m,n,d,wtm,wtn);
```

Algebraic complexity theory

"Compute the commuting left and right action of a tensor product of quantum enveloping algebras on a given vector space"

Summary

- Programming is a recurrent thread in all aspects of computer science
- Teaching programming effectively requires well-motivated problems to solve
- “Realistic” programming achieves operational understanding that must precede abstraction of concepts