Intelligent Patterning

and problem solving

Tom Carter

CSU Stanislaus

December 27, 2018

Brief overview

General problem solving

- **Understanding the problem**
  1. Problem context and statement of the problem
  2. Solving the right problem (ill-posed and ill-conditioned problems)
  3. Preconceptions
  4. Language and restating the problem
- **The role of experience**
  1. Similar problems and analogy
  2. Appropriate tools
  3. Specific experience

- **Three basic methods**
  1. Plug and grind
  2. Guess and prove
  3. Look it up
- **Hypothesis generation and testing**
  1. Flexibility and freedom — willingness to try and fail
  2. Recognizing blind alleys, and the value of exploring
  3. Appropriate hypotheses
  4. Lateral thinking

- **Recognizing solutions**
  1. "A" solution vs. "the" solution
  2. Useful solutions
  3. When a "solution" solves an un-posed, but more significant problem

Pattern recognition

- Images ("visual patterns") vs. "syntactic" patterns
- Symbols as patterns, and symbols as pattern labels
- Patterns of symbols
- Hierarchies of patterns, and symbols as tools for recognizing patterns
- Pattern manipulation
- Learning to recognize patterns, and pattern recognition as learning

Pattern recognition examples

- What number comes next in the sequence?
  1, 1, 2, 3, 5, 8, 13, . . .

Pattern recognition examples

- What number comes next in the sequence?
  1, 1, 2, 3, 5, 8, 13, . . .
- What number comes next in the sequence?
  8, 5, 4, 9, 1, 7, 6, 3, . . .

Pattern recognition examples

- What number comes next in the sequence?
  1, 1, 2, 3, 5, 8, 13, . . .
- What number comes next in the sequence?
  8, 5, 4, 9, 1, 7, 6, 3, . . .
- What letter comes next in the sequence?
  E, T, A, O, I, N, S, H, . . .

Pattern recognition examples

- What number comes next in the sequence?
  1, 1, 2, 3, 5, 8, 13, . . .
- What number comes next in the sequence?
  8, 5, 4, 9, 1, 7, 6, 3, . . .
- What letter comes next in the sequence?
  E, T, A, O, I, N, S, H, . . .
- In which row does Z go?
  A, E, F, H, I, K, L, M, N, T, V, W, X, Y
  B, C, D, G, J, O, P, Q, R, S, U

Pattern recognition examples

- What number comes next in the sequence?
  1, 1, 2, 3, 5, 8, 13, ...
- What number comes next in the sequence?
  8, 5, 4, 9, 1, 7, 6, 3, ...
- What letter comes next in the sequence?
  E, T, A, O, I, N, S, H, ...
- In which row does Z go?
  A, E, F, H, I, K, L, M, N, T, V, W, X, Y
  B, C, D, G, J, O, P, Q, R, S, U
- What letter comes next in the sequence?
  W, L, C, N, I, T, ...

Symbols and signs

- The utility and power of symbols
- Choosing symbols, naming and pointing
- Symbols as “chunking” tools
- When to use symbols
  1. The importance of anonymity (e.g., the lambda calculus)
  2. Place holders (variables)
  3. Temporary and tentative symbols
- Signs, symbols, content and meaning

Intelligent patterning

- Creativity and Art
  1. Knowing when to pattern
  2. Symbol attachment and creation; patterns/symbols as revealers and concealers
  3. Levels of patterning
- Multiple patterns and selection
  $(x-1)(x-2)(x-3)-6$
  $x^3-6x^2+11x-12$
  $(x-4)(x^2-2x+3)$
- Adaptive pattern recognition
- Are the patterns really there?

Some history

- Physics
- Philosophy (theory of knowledge)
- Mathematics
  1. Matrix manipulation
  2. Topology
  3. Algebra
  4. Lie groups
  5. Manifolds and relativity theory
  6. Algebraic topology

Making things look right:

Consider this piece of mathematics (here, and the next page).
How do we get this typeset? See the following page for LATEX . . .

We have the map $b_n : \Sigma^2 U(n) \to SU(n+1)$
given by

$$b_n(g, r, s) = [i(g), v_n(r, s)]$$

where $i(g)$ is the inclusion, $[g, h] = ghg^{-1}h^{-1}$
and

$v_n(r,s) =$

$$\begin{bmatrix} \alpha & 0 & 0 & \cdots & 0 & \beta(-\overline{\alpha})^0 \\ \beta(-\overline{\alpha})^0\overline{\beta} & \alpha & 0 & \cdots & 0 & \beta(-\overline{\alpha})^1 \\ \beta(-\overline{\alpha})^1\overline{\beta} & \beta(-\overline{\alpha})^0\overline{\beta} & \alpha & \cdots & 0 & \beta(-\overline{\alpha})^2 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \beta(-\overline{\alpha})^{n-1}\overline{\beta} & \beta(-\overline{\alpha})^{n-2}\overline{\beta} & \cdots & \cdots & \alpha & \beta(-\overline{\alpha})^n \\ -(-\overline{\alpha})^n\overline{\beta} & -(-\overline{\alpha})^{n-1}\overline{\beta} & \cdots & \cdots & -(-\overline{\alpha})^0\overline{\beta} & -(-\overline{\alpha})^n \end{bmatrix}$$

where

$$\alpha = \alpha(r,s) = \cos(\pi r) + i\sin(\pi r)\cos(\pi s)$$

$$\beta = \beta(r,s) = i\sin(\pi r)\sin(\pi s)$$

```
We have the map
    \$ b_n: \Sigma^2U(n) \rightarrow SU(n+1) \$ \newline
given by
    \[ b_n(g, r, s) = \left[ i(g), v_n(r, s) \right] \]
where $i(g)$ is the inclusion,
    $\left[g, h\right] = ghg^{-1}h^{-1}$ \newline
and
$ v_n(r,s) = $
\[ \left[ \begin{array}{cccccc}
    \alpha & 0 & 0 & \cdots & 0 & \beta (-\overline{\alpha})^0 \\
    \beta (-\overline{\alpha})^0\overline{\beta} &
      \alpha  &  0  & \cdots & 0 &
      \beta (-\overline{\alpha})^1 \\
    \beta (-\overline{\alpha})^1\overline{\beta} &
      \beta (-\overline{\alpha})^0\overline{\beta} &
      \alpha & \cdots & 0 & \beta (-\overline{\alpha})^2 \\
    \vdots & \vdots & \vdots &  & \vdots & \vdots \\
    \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
    \vdots & \vdots & \vdots &  & \vdots & \vdots \\
    \beta (-\overline{\alpha})^{n-1}\overline{\beta} &
      \beta (-\overline{\alpha})^{n-2}\overline{\beta} &
      \cdots &  \cdots & \alpha &
      \beta (-\overline{\alpha})^n \\
    -(-\overline{\alpha})^n\overline{\beta} &
      -(-\overline{\alpha})^{n-1}\overline{\beta} &
      \cdots &  \cdots & -(-\overline{\alpha})^0
      \overline{\beta} & -(-\overline{\alpha})^n \\
    \end{array} \right]
\]
where
\[ \alpha = \alpha(r,s) =
    \cos(\pi r) + i \sin(\pi r)\cos(\pi s) \]
\[ \beta = \beta(r,s) = i \sin(\pi r)\sin(\pi s) \]
```

What’s wrong in computing today

- Not enough resolution on displays (seems mostly solved . . . :-)
- Not enough processing power and memory
- Not enough parallelism
- Software tools are (largely) “flat” and sequential
  rather than hierarchical

The intelligent mathematical assistant

- Adaptive symbolic input and output
- Strong basic skills (all of arithmetic through college calculus and elementary discrete structures)
- First order logic capabilities
- Adaptive "patterning" and "symboling"
- Elementary hypothesis generation and testing