Thinking in APL: Array-oriented Solutions (Part 1)

Richard Park

Thinking in APL: Array-oriented Solutions

Tool of thought

Language & thought

Primitives

Idioms

Thinking in APL: ***Array-oriented Solutions***

Array as the unit

Direct expression

Techniques

Heuristics

This Webinar

Food for thought

Secret sauce

Notation as a Tool of Thought

Iverson, K.E., 2007.

In ACM Turing award lectures (p. 1979).

Notation as a Tool of Thought

Ease of expressing constructs arising in problems.

Suggestivity.

Ability to subordinate detail.

Economy.

Amenability to formal proofs.

Notation as a Tool of Thought

Design Patterns vs Anti pattern in APL by Aaron W Hsu at FnConf17

https://www.youtube.com/watch?v=v7Mt0GYHU9A

Language as a Tool of Thought

Expression

Suggestivity

Subordination of detail

Economy

Economy

A Conversation with Arthur Whitney (ACM 2009)

Brian Cantrill & Arthur Whitney

AW: … we can remember seven things.

BC: Right. People are able to retain a seven-digit phone number, but it drops off quickly at eight, nine, ten digits.

AW: If you‘re Cantonese, then it’s ten. I have a very good friend, Roger Hui, who implements J. He was born in Hong Kong but grew up in Edmonton as I did. One day I asked him, “Roger, do you do math in English or Cantonese?” He smiled at me and said, “I do it in Cantonese because it‘s faster and it’s completely regular.”

APL Thinking?

The thought process of someone using APL

- Primitive functions and operators
- Translating natural language algorithm descriptions
- Translating pseudo code
- Translating code from another programming language
- Translating mathematical formulae
- Specific techniques
- Problem solving heuristics

APL Thinking?

The thought process of someone using APL

- Primitive functions and operators
- Translating natural language algorithm descriptions
- Translating pseudo code
- Translating code from another programming language
- Translating mathematical formulae
- **Specific techniques**
- **Problem solving heuristics**

Pragmatism: Array-oriented Solutions?

Array as a unit

Array as a unit

Example: *Counting elements*

```
 ∇  t←Count array;element
[1]    t←0
[2]    :For element :In array
[3]        t+←1
[4]    :EndFor
    ∇
```

Array as a unit

Example: *Counting elements*

```
{⍺←0 ⋄ 0=⍴⍵:⍺ ⋄ (⍺+1)∇1↓⍵}
{+/⍵=⍵}
```

Array as a unit

Example: *Counting elements*

```
{⍺←0 ⋄ 0=⍴⍵:⍺ ⋄ (⍺+1)∇1↓⍵}
{+/⍵=⍵}      +/=⍨
             ≢
```

Array as a unit

Example: *Selection*

```
 ∇  vowels←JustVowels word
[1]    vowels←''
[2]    :For letter :In word
[3]        :If letter∊'aeiou'
[4]            vowels,←letter
[5]        :EndIf
[6]    :EndFor
    ∇
```

Array as a unit

Example: *Selection*

```
'aeiou'{(⍵∊⍺)/⍵}word
```

Array-oriented Solutions

Metzger, R.C., 1981. APL thinking finding array-oriented solutions. *ACM SIGAPL APL Quote Quad*, *12*(1), pp.212-218.

Eisenberg, M. and Peelle, H.A., 1987. APL thinking: examples. *ACM SIGAPL APL Quote Quad*, *17*(4), pp.433-440.

Knowing and using

Primitives

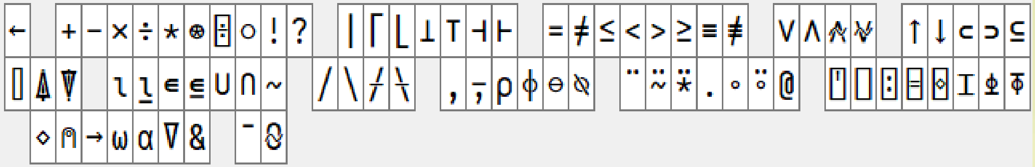
Idioms

Techniques

Heuristics

Primitives

Language bar

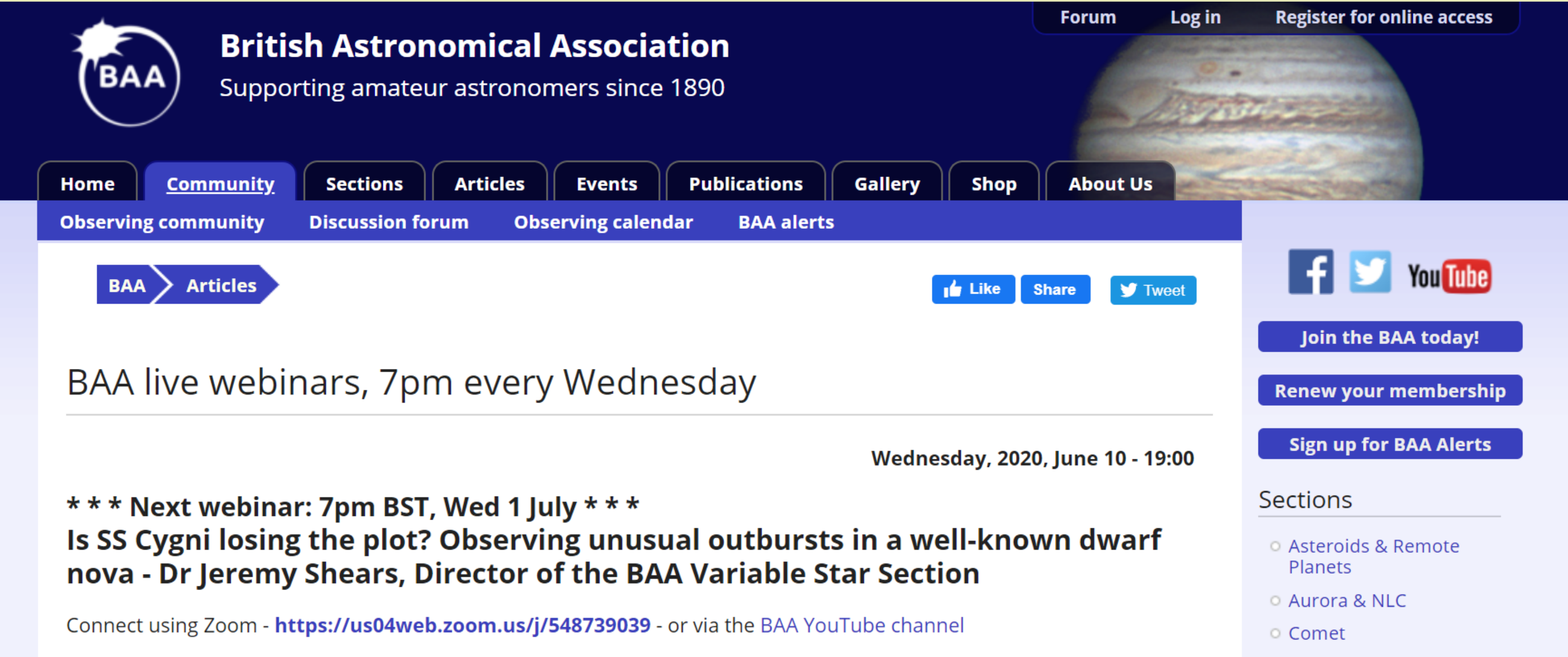


Idioms

Jul 2nd 16:00 BST aplcart.info turns 1

Adám Brudzewsky demonstrates the various features of APLcart, the largest-ever collection of short APL phrases.

Search: BAA webinar schedule 2020



British Astronomical Association

Supporting amateur astronomers since 1890

BAA live webinars, 7pm every Wednesday

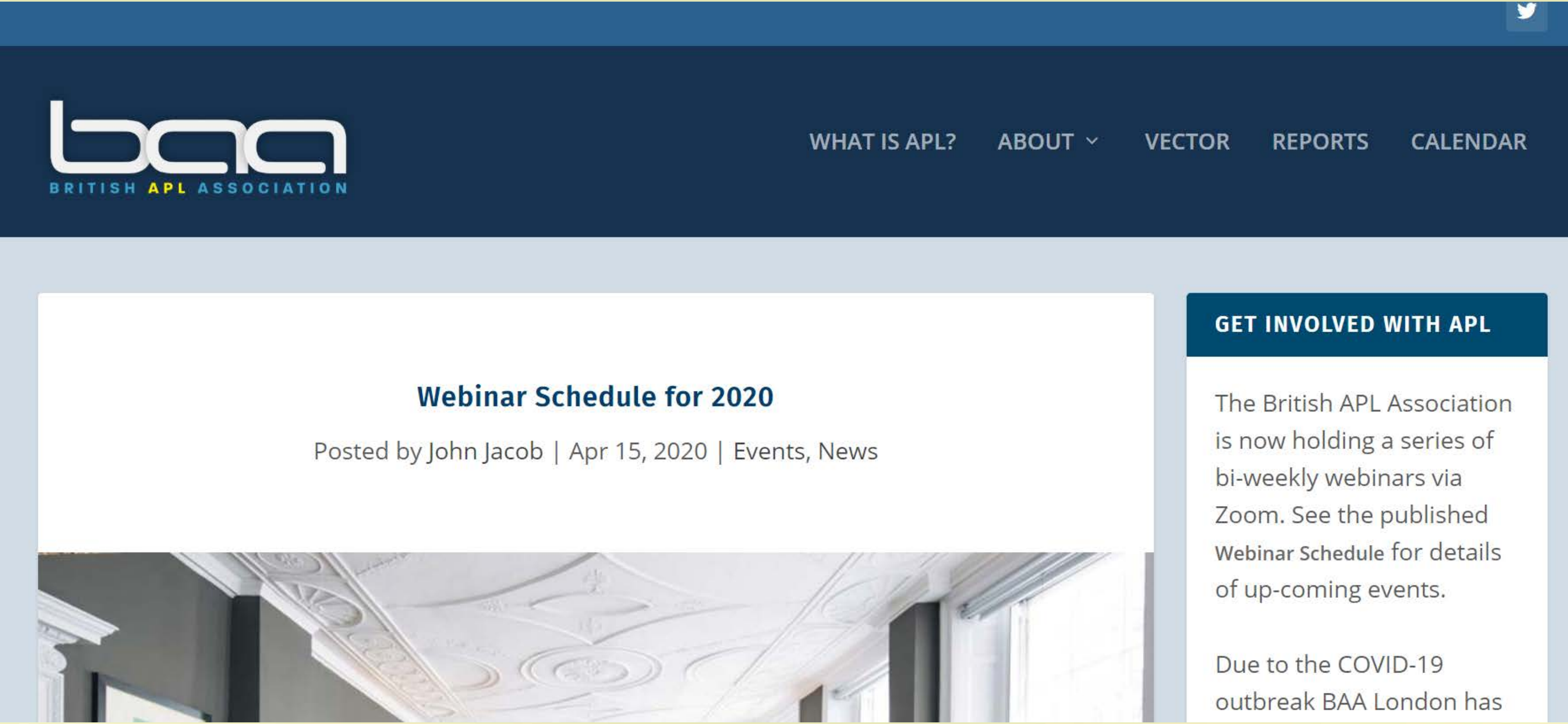
Wednesday, 2020, June 10 - 19:00

*** * * Next webinar: 7pm BST, Wed 1 July * * ***
Is SS Cygni losing the plot? Observing unusual outbursts in a well-known dwarf nova - Dr Jeremy Shears, Director of the BAA Variable Star Section

Connect using Zoom - **https://us04web.zoom.us/j/548739039** - or via the BAA YouTube channel

Sections

- Asteroids & Remote Planets
- Aurora & NLC
- Comet



Webinar Schedule for 2020

Posted by John Jacob | Apr 15, 2020 | Events, News

GET INVOLVED WITH APL

The British APL Association is now holding a series of bi-weekly webinars via Zoom. See the published **Webinar Schedule** for details of up-coming events.

Due to the COVID-19 outbreak BAA London has

Techniques

```
{f⌿  α  g  ω}
{(α  f  ω)⌿ω}
```

Heuristics

Metzger, R.C., 1981.

APL thinking finding array-oriented solutions.

ACM SIGAPL APL Quote Quad, *12*(1), pp.212-218.

Heuristics

Metzger, R.C., 1981.

APL thinking finding array-oriented solutions.

1) Value First, Then Shape;
2) Shape First, Then Value;
3) Data Transformation;
4) Loop First;
5) Think Big;
6) Function Listing;
7) Synonym Search.

Value First

Example: To

Value First

```
      3{(⍺-1)↓⍳⍵}7
3 4 5 6 7
```

Value First

Filtering

```
        3{(α≤i)/i←⍳ω}7
3 4 5 6 7
```

Shape First

```
      3{(⍺-1)+⍳1+⍵-⍺}7
3 4 5 6 7
```

Value First

```
Miota←{
        max←⌈/⍵
        i←⍳max
        n←((⍴⍵),max)⍴i
        (,⍵∘.≥i)/,n
  }
```

```
      MIota 4 2 3
1 2 3 4 1 2 1 2 3
```

Shape First

```
Miota2←{
        i←(+/⍵)⍴1
        i[1++\¯1↓⍵]←1-¯1↓⍵
        +\i
  }
```

```
      MIota 4 2 3
1 2 3 4 1 2 1 2 3
```

Shape First

```
Miota3←{
       i←(+/⍵)⍴1
       +\(1-¯1↓⍵)@(1++\¯1↓⍵)⊢i
  }
```

```
      MIota 4 2 3
1 2 3 4 1 2 1 2 3
```

Language as a Tool of Thought

“the computer language you use influences how you understand and solve problems”

- Metzger, R.C., 1981. APL thinking finding array-oriented solutions.

Linguistic Determinism

"the language you use influences how you understand" – me just now

Majid, A., Bowerman, M., Kita, S., Haun, D.B. and Levinson, S.C., 2004. **Can language restructure cognition? The case for space.** *Trends in cognitive sciences*, *8*(3), pp.108-114.

Linguistic Determinism

Search: Sapir-Whorf Hypothesis

Radiolab Words

Majid, A., Bowerman, M., Kita, S., Haun, D.B. and Levinson, S.C., 2004. **Can language restructure cognition? The case for space.** *Trends in cognitive sciences*, *8*(3), pp.108-114.

APL Thinking

- Metzger, R.C., 1981. APL thinking finding array-oriented solutions.
- Eisenberg, M. and Peelle, H.A., 1987. APL thinking: examples. *ACM SIGAPL APL Quote Quad*, *17*(4), pp.433-440.
- Eisenberg, M. and Peelle, H.A., 1990. A survey “APL thinking”. *ACM SIGAPL APL Quote Quad*, *21*(2), pp.5-8.
- Eisenberg, M. and Peelle, H.A., 1983. APL learning bugs. *ACM SIGAPL APL Quote Quad*, *13*(3), pp.11-16.
- Peelle, H.A. and Eisenberg, M., 1985, May. APL teaching bugs. In *Proceedings of the international conference on APL: APL and the future* (pp. 86-93).
- Eisenberg, M. and Peelle, H., 1989, August. APL problem-solving (tutorial session) a tutorial. In *Proceedings of the ACM/SIGAPL conference on APL as a tool of thought (session tutorials)* (pp. 1-30).
- Polivka, R.P., 1984, June. The impact of APL2 on teaching APL. In *Proceedings of the international conference on APL* (pp. 263-269).

Array-oriented Solutions

Primitives

Idioms

Techniques

Heuristics

Next Week

Jul 2nd 16:00 BST aplcart.info turns 1

Adám Brudzewsky demonstrates the various features of APLcart, the largest-ever collection of short APL phrases.

Search: BAA webinar schedule 2020

Next Dyalog Webinar

Jul 9th 16:00 BST

Adám presents

Language Features of Dyalog version 18.0 in Depth (part 3)