

CS 42I Lecture 15

- ▶ Today's class: APL
 - ▶ Functional programming – “no side effects”

Functional Programming

- ▶ “The assignment statement splits programming into two worlds. The first world comprises the right sides of assignment statements. This is an orderly world of expressions, a world that has useful algebraic properties.... It is the world in which most useful computation takes place.
- ▶ “The second world... is the world of statements. ... This world of statements is a disorderly one, with few useful mathematical properties.”

John Backus (creator of Fortran), “Can Programming be liberated from the von Neumann Style? A Functional Style and its Algebra of Programs.”
Turing Award lecture, 1977.

APL

- ▶ Computations on matrices using operators that have matrix arguments.
- ▶ Ken Iverson – “A Programming Language” – 1960
- ▶ Defined a set of operators on matrices, plus a typeface for those operators, and built terminals

APL operations

- ▶ **Binary operations on numbers extended naturally to matrices**
 - Comparison and boolean ops treated as arithmetic
- ▶ **Reduction operations: $+ /$, $\times /$, $\wedge /$, ...**
 - For vectors, put operator between every element
 - For matrices, reduce each row
- ▶ **Compression: B / V**
 - selects elements (or rows) of V where $B = 1$

APL font

ABCDEFGHIJKLMNOPQRSTUVWXYZ

0123456789

^ % ◡ < > = ¼ ½ ¾ ^ / ÷ „ * × + -

APL examples

- ▶ $1 + M$
- ▶ $(+ / V) \div n$
- ▶ $(+ / V) \div \frac{1}{2}V$
- ▶ $(((V \div 2) \times 2) = V) / V$

APL examples

▶ `prime n = ^/(0~n%(1+1/4(n-2)))`

APL examples

- ▶ Subscripting: $V[V']$ – elements of V in positions given by V' .

reverse $V = V[1 + (\frac{1}{2}V) - \frac{1}{4}1\frac{1}{2}V]$

APL examples

- ▶ Dyadic $\frac{1}{2}$ – “restructure”
- ▶ $V\frac{1}{2}A$ returns a value with shape V , values drawn from A

APL examples

- ▶ „ assignment
- ▶ ³ transpose

$$({}^3M) = M, (2\frac{1}{2}n) \frac{1}{2}\frac{1}{4}n$$

APL examples

```
let zero = newint 0;;
```

```
let four = newint 4;;
```

```
let a = rho(newveci [2;3]) (indx (newint 6));;
```

```
let v = newveci [2;4;6];;
```

```
let c = newveci [1;0];;
```

```
let d = newveci [1;0;1];;
```

```
a * @ a
```

```
v - @ one
```

```
a > @ four
```

```
!+v
```

APL examples

`maxR a`

`d % v`

`c % a`

`shape a`

`ravel a`

`rho (shape a) v`

`rho (shape v) c`

`a ^@ c`

APL examples

indx (newint 5)

trans a

v @@ (indx two)

a @@ one

(trans a) @@ (indx two)

APL examples

```
let incr a = a +@ (newint 1);;
```

```
let fac n = !* (indx n);;
```

```
let avg v = (!+v) /@ (shape v);;
```

```
let reverse v –
```

```
  let sz = (shape v) @@ one
```

```
  in v @@ (incr (sz -@ (indx sz)));;
```

```
let prime n = !& (zero <>@ (n % @ (incr  
  (indx (n -@ two)))));;
```