Introduction to Array Programming

Rob Moore

Notation as a tool of thought

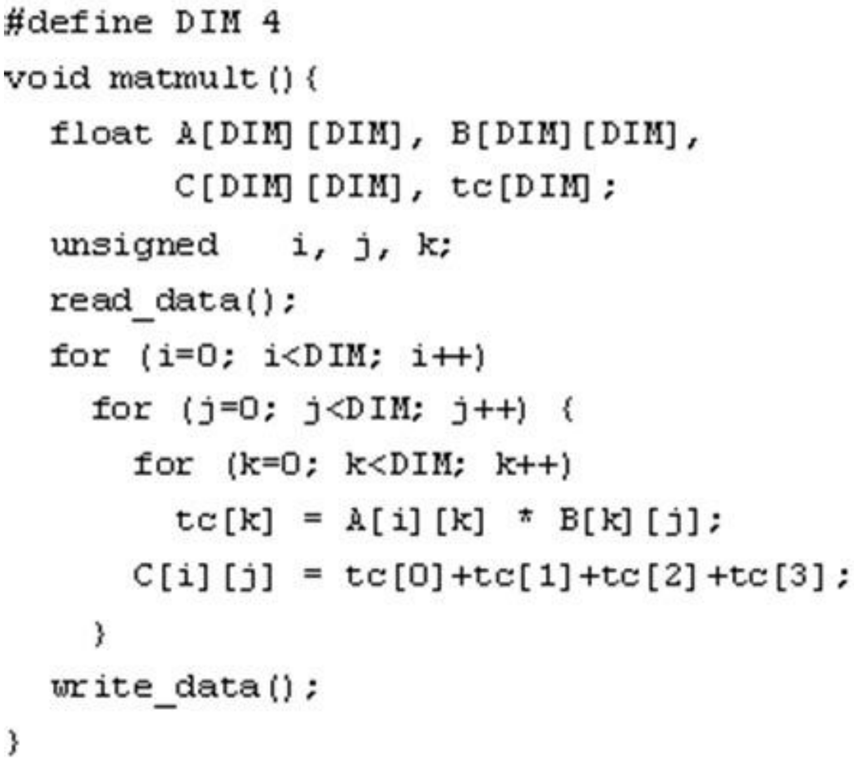
0

Σ

!

∞

π



```
#define DIM 4
void matmult(){
  float A[DIM][DIM], B[DIM][DIM],
        C[DIM][DIM], tc[DIM];
  unsigned   i, j, k;
  read_data();
  for (i=0; i<DIM; i++)
    for (j=0; j<DIM; j++) {
      for (k=0; k<DIM; k++)
        tc[k] = A[i][k] * B[k][j];
      C[i][j] = tc[0]+tc[1]+tc[2]+tc[3];
    }
  write_data();
}
```

$$c_{ij} = \sum_{k=1}^{m} a_{ik} b_{kj}$$

```
c =: a (+/ . *) b
```

Nouns and Verbs

- Nouns are data
- Verbs are ‘doing words’

Nouns and Verbs

- Nouns are data
- Verbs are ‘doing words’
    - Infix

Nouns and Verbs

- Nouns are data
- Verbs are ‘doing words’
    - Infix
    - Max of 2 inputs. Left and right.

Nouns and Verbs

- Nouns are data
- Verbs are ‘doing words’
    - Infix
    - Max of 2 inputs. Left and right.
    - 1 input: Monadic

Nouns and Verbs

- Nouns are data
- Verbs are ‘doing words’
    - Infix
    - Max of 2 inputs. Left and right.
    - 1 input: Monadic
    - 2 inputs: Dyadic

Scalars and Vectors

Scalars and Vectors

- Scalars: 1 and ‘a’ (aka atoms)

Scalars and Vectors

- Scalars: 1 and ‘a’ (aka atoms)
- Vectors: 10 11 12 and ‘hello world’

Scalars and Vectors

- Scalars: 1 and ‘a’ (aka atoms)
- Vectors: 10 11 12 and ‘hello world’

Verb rank

Traffic data demo

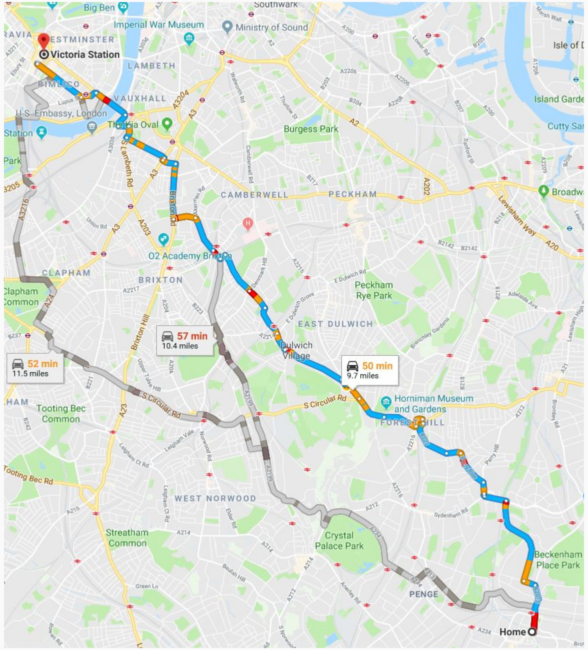
Traffic data demo

Based on the available traffic data, how can I minimise the number of bicycles my shuttle bus service encounters daily?

Traffic data demo

Data we have:

- Average annual daily flow of bicycles along all major roads in London.
- The major roads along the routes.



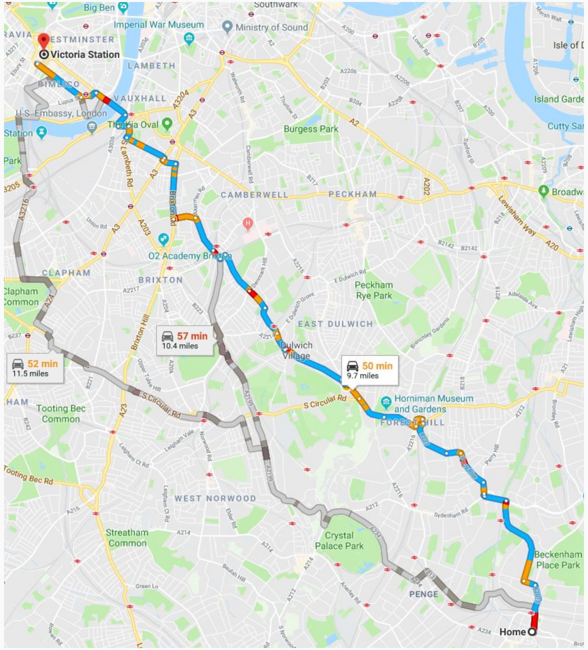
Traffic data demo

Data we have:

- Average annual daily flow of bicycles along all major roads in London.
- The major roads along the routes.

Question we want answered:

- Along which route will my shuttle bus service encounter the fewest bicycles?



Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
```

Traffic data demo

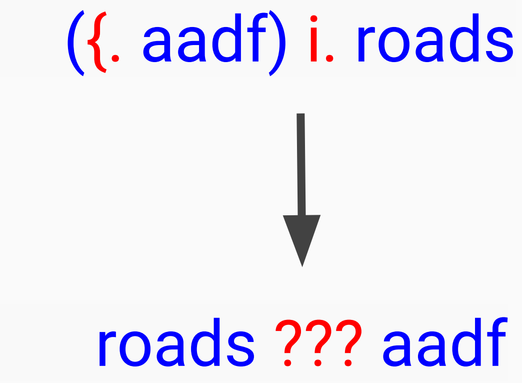
```
+/ (({. aadf) i. roads) { ({: aadf)
```

↓

```
roads ??? aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
```



```
({. aadf) i. roads
```

```
roads ??? aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
```

```
({. aadf) i. roads
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
```

```
  ({. aadf) i. roads
⇔ roads i.~ ({. aadf)
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
```

```
   ({. aadf) i. roads
⇔ roads i.~ ({. aadf)
⇔ roads (i.~ {.) aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
+/ (roads (i.~ {.) aadf) { ({: aadf)
```

```
  ({. aadf) i. roads
⇔ roads i.~ ({. aadf)
⇔ roads (i.~ {.) aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
+/ (roads (i.~ {.) aadf) { ({: aadf)
```

(x f y) g (x h y) ⇔ x (f g h) y

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
+/ (roads (i.~ {.) aadf) { ({: aadf)
```

```
(x f y) g (x h y) ⇔ x (f g h) y
```

```
{: aadf
```

↓

```
roads ??? aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
+/ (roads (i.~ {.) aadf) { ({: aadf)
```

```
(x f y) g (x h y) ⇔ x (f g h) y

roads ] aadf ⇔aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
+/ (roads (i.~ {.) aadf) { ({: aadf)
```

```
(x f y) g (x h y) ⇔ x (f g h) y

roads     ] aadf ⇔    aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
+/ (roads (i.~ {.) aadf) { ({: aadf)
```

```
(x f y) g (x h y) ⇔ x (f g h) y
```

```
roads     ] aadf ⇔    aadf
roads {:@] aadf ⇔ {: aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
+/ (roads (i.~ {.) aadf) { ({: aadf)
+/ roads ((i.~ {.) { {:@]) aadf
```

```
(x f y) g (x h y) ⇔ x (f g h) y

roads     ] aadf ⇔    aadf
roads {:@] aadf ⇔ {: aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
+/ (roads (i.~ {.) aadf) { ({: aadf)
+/ roads ((i.~ {.) { {:@]) aadf
```

Traffic data demo

```
+/ (({. aadf) i. roads) { ({: aadf)
+/ (roads (i.~ {.) aadf) { ({: aadf)
+/ roads ((i.~ {.) { {:@]) aadf
roads +/@:((i.~ {.) { {:@]) aadf
```

Thanks!

robmoore121@gmail.com
https://github.com/robknows