Simple program that returns three values:

$$\mathrm{PRGM}(x) := \begin{Vmatrix} a \leftarrow x \\ b \leftarrow x^2 \\ c \leftarrow \sqrt{x} \\ \begin{bmatrix} a & b & c \end{bmatrix} \end{Vmatrix}$$

$$\mathbf{ORIGIN} \equiv 1$$

Sample data to feed PRGM:

$$Data1 := \begin{bmatrix} 2 \\ 3 \\ 5 \\ 7 \\ 11 \\ 13 \\ 17 \\ 19 \end{bmatrix}$$

$$i1 := \mathbf{ORIGIN} .. \mathrm{last}(Data1)$$

$$Data2 := \begin{bmatrix} 23 \\ 29 \\ 31 \\ 37 \\ 39 \end{bmatrix}$$

$$i2 := \mathbf{ORIGIN} .. \mathrm{last}(Data2)$$

Results from using data arrays as input objects:

$$P1a := \mathrm{PRGM}(Data1) = \begin{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 5 \\ 7 \\ 11 \\ 13 \\ 17 \\ 19 \end{bmatrix} & \begin{bmatrix} 4 \\ 9 \\ 25 \\ 49 \\ 121 \\ 169 \\ 289 \\ 361 \end{bmatrix} & \begin{bmatrix} 1.414 \\ 1.732 \\ 2.236 \\ 2.646 \\ 3.317 \\ 3.606 \\ 4.123 \\ 4.359 \end{bmatrix} \end{bmatrix}$$

$$P2a := \mathrm{PRGM}(Data2) = \begin{bmatrix} \begin{bmatrix} 23 \\ 29 \\ 31 \\ 37 \\ 39 \end{bmatrix} & \begin{bmatrix} 529 \\ 841 \\ 961 \\ 1369 \\ 1521 \end{bmatrix} & \begin{bmatrix} 4.796 \\ 5.385 \\ 5.568 \\ 6.083 \\ 6.245 \end{bmatrix} \end{bmatrix}$$

Results from using elements of arrays as input objects:

$$P1e_{i1} := \mathrm{PRGM}\left(Data1_{i1}\right) = \begin{bmatrix} \begin{bmatrix} 2 & 4 & 1.414 \end{bmatrix} \\ \begin{bmatrix} 3 & 9 & 1.732 \end{bmatrix} \\ \begin{bmatrix} 5 & 25 & 2.236 \end{bmatrix} \\ \begin{bmatrix} 7 & 49 & 2.646 \end{bmatrix} \\ \begin{bmatrix} 11 & 121 & 3.317 \end{bmatrix} \\ \begin{bmatrix} 13 & 169 & 3.606 \end{bmatrix} \\ \begin{bmatrix} 17 & 289 & 4.123 \end{bmatrix} \\ \begin{bmatrix} 19 & 361 & 4.359 \end{bmatrix} \end{bmatrix}$$

$$P2e_{i2} := \mathrm{PRGM}\left(Data2_{i2}\right) = \begin{bmatrix} \begin{bmatrix} 23 & 529 & 4.796 \end{bmatrix} \\ \begin{bmatrix} 29 & 841 & 5.385 \end{bmatrix} \\ \begin{bmatrix} 31 & 961 & 5.568 \end{bmatrix} \\ \begin{bmatrix} 37 & 1369 & 6.083 \end{bmatrix} \\ \begin{bmatrix} 39 & 1521 & 6.245 \end{bmatrix} \end{bmatrix}$$

$$\overrightarrow{\mathrm{PRGM}(Data2)} = \begin{bmatrix} \begin{bmatrix} 23 & 529 & 4.796 \end{bmatrix} \\ \begin{bmatrix} 29 & 841 & 5.385 \end{bmatrix} \\ \begin{bmatrix} 31 & 961 & 5.568 \end{bmatrix} \\ \begin{bmatrix} 37 & 1369 & 6.083 \end{bmatrix} \\ \begin{bmatrix} 39 & 1521 & 6.245 \end{bmatrix} \end{bmatrix}$$

What I really want...single arrays without subarrays:

$$P1 = ????? = \begin{bmatrix} 2 & 4 & 1.414 \\ 3 & 9 & 1.732 \\ 5 & 25 & 2.236 \\ 7 & 49 & 2.646 \\ 11 & 121 & 3.317 \\ 13 & 169 & 3.606 \\ 17 & 289 & 4.123 \\ 19 & 361 & 4.359 \end{bmatrix}$$

$$P2 = ????? = \begin{bmatrix} 23 & 529 & 4.796 \\ 29 & 841 & 5.385 \\ 31 & 961 & 5.568 \\ 37 & 1369 & 6.083 \\ 39 & 1521 & 6.245 \end{bmatrix}$$

$$P1^{\langle i1 \rangle} := \overrightarrow{\mathrm{PRGM}\left(Data1_{i1}\right)}$$

$$P1 = \begin{bmatrix} 2 & 4 & 1.414 \\ 3 & 9 & 1.732 \\ 5 & 25 & 2.236 \\ 7 & 49 & 2.646 \\ 11 & 121 & 3.317 \\ 13 & 169 & 3.606 \\ 17 & 289 & 4.123 \\ 19 & 361 & 4.359 \end{bmatrix}$$

$$P2^{\langle i2 \rangle} := \overrightarrow{\mathrm{PRGM}\left(Data2_{i2}\right)}$$

$$P2 = \begin{bmatrix} 23 & 529 & 4.796 \\ 29 & 841 & 5.385 \\ 31 & 961 & 5.568 \\ 37 & 1369 & 6.083 \\ 39 & 1521 & 6.245 \end{bmatrix}$$

```
unnest(V) :=  O ← ORIGIN
              if min(rows(V), cols(V)) ≠ 1
                  return V
              if min(rows(V_{O,O}), cols(V_{O,O})) ≠ 1
                  return V
              if (min(→cols(V)) ≠ max(→cols(V))) ∨ (min(→rows(V)) ≠ max(→rows(V)))
                  return V
              if cols(V) > 1
                  V ← V^T
              if cols(V_O) = 1
                  for i ∈ O .. last(V)
                      M^⟨i⟩ ← V_i
              else
                  for i ∈ O .. last(V)
                      M^⟨i⟩ ← V_i
              M
```

$$P1a = \begin{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 5 \\ 7 \\ 11 \\ 13 \\ 17 \\ 19 \end{bmatrix} & \begin{bmatrix} 4 \\ 9 \\ 25 \\ 49 \\ 121 \\ 169 \\ 289 \\ 361 \end{bmatrix} & \begin{bmatrix} 1.414 \\ 1.732 \\ 2.236 \\ 2.646 \\ 3.317 \\ 3.606 \\ 4.123 \\ 4.359 \end{bmatrix} \end{bmatrix} \qquad unnest(P1a) = \begin{bmatrix} 2 & 4 & 1.414 \\ 3 & 9 & 1.732 \\ 5 & 25 & 2.236 \\ 7 & 49 & 2.646 \\ 11 & 121 & 3.317 \\ 13 & 169 & 3.606 \\ 17 & 289 & 4.123 \\ 19 & 361 & 4.359 \end{bmatrix}$$

$$P2e = \begin{bmatrix} \begin{bmatrix} 23 & 529 & 4.796 \end{bmatrix} \\ \begin{bmatrix} 29 & 841 & 5.385 \end{bmatrix} \\ \begin{bmatrix} 31 & 961 & 5.568 \end{bmatrix} \\ \begin{bmatrix} 37 & 1369 & 6.083 \end{bmatrix} \\ \begin{bmatrix} 39 & 1521 & 6.245 \end{bmatrix} \end{bmatrix} \qquad unnest(P2e) = \begin{bmatrix} 23 & 529 & 4.796 \\ 29 & 841 & 5.385 \\ 31 & 961 & 5.568 \\ 37 & 1369 & 6.083 \\ 39 & 1521 & 6.245 \end{bmatrix}$$

$$P1e^{T} = [[2\ \ 4\ \ 1.414]\ \ [3\ \ 9\ \ 1.732]\ \ [5\ \ 25\ \ 2.236]\ \ [7\ \ 49\ \ 2.646]\ \ [11\ \ 121\ \ 3.317]\ \ [13\ \ 169\ \ 3.606]\ \ [17\ \ 289\ \ 4.1$$

$$P2a^{T} = \begin{bmatrix} \begin{bmatrix} 23 \\ 29 \\ 31 \\ 37 \\ 39 \end{bmatrix} \\ \begin{bmatrix} 529 \\ 841 \\ 961 \\ 1369 \\ 1521 \end{bmatrix} \\ \begin{bmatrix} 4.796 \\ 5.385 \\ 5.568 \\ 6.083 \\ 6.245 \end{bmatrix} \end{bmatrix}$$

$$unnest\left(P1e^{T}\right) = \begin{bmatrix} 2 & 4 & 1.414 \\ 3 & 9 & 1.732 \\ 5 & 25 & 2.236 \\ 7 & 49 & 2.646 \\ 11 & 121 & 3.317 \\ 13 & 169 & 3.606 \\ 17 & 289 & 4.123 \\ 19 & 361 & 4.359 \end{bmatrix}$$

The routine is written on purpose so that it returns a matrix in those cases and not a row- or column vector

$$unnest\left(P2a^{T}\right) = \begin{bmatrix} 23 & 529 & 4.796 \\ 29 & 841 & 5.385 \\ 31 & 961 & 5.568 \\ 37 & 1369 & 6.083 \\ 39 & 1521 & 6.245 \end{bmatrix}$$

$$\text{PRGM1}(x) := \left\| \begin{array}{l} a \leftarrow |x| \\ b \leftarrow x \cdot x \\ c \leftarrow \sqrt{x} \\ \begin{bmatrix} a & b & c \end{bmatrix} \end{array} \right.$$

$$\text{PRGM2}(x) := \left\| \begin{array}{l} a \leftarrow |x| \\ b \leftarrow x \cdot x \\ c \leftarrow \sqrt{x} \\ \begin{bmatrix} a & b & c \end{bmatrix}^{\mathrm{T}} \end{array} \right.$$

$$Data := \begin{bmatrix} -4 \\ -1 \\ 0 \\ 9 \end{bmatrix}$$

$$\overrightarrow{\text{PRGM1}(Data)} = \begin{bmatrix} \begin{bmatrix} 4 & 16 & 2i \end{bmatrix} \\ \begin{bmatrix} 1 & 1 & 1i \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 9 & 81 & 3 \end{bmatrix} \end{bmatrix}$$

$$\overrightarrow{\text{PRGM2}(Data)} = \begin{bmatrix} \begin{bmatrix} 4 \\ 16 \\ 2i \end{bmatrix} \\ \begin{bmatrix} 1 \\ 1 \\ 1i \end{bmatrix} \\ \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ \begin{bmatrix} 9 \\ 81 \\ 3 \end{bmatrix} \end{bmatrix}$$

$$P1 := unnest\left(\overrightarrow{\text{PRGM1}(Data)}\right) = \begin{bmatrix} 4 & 16 & 2i \\ 1 & 1 & 1i \\ 0 & 0 & 0 \\ 9 & 81 & 3 \end{bmatrix}$$

$$P2 := unnest\left(\overrightarrow{\text{PRGM2}(Data)}\right) = \begin{bmatrix} 4 & 1 & 0 & 9 \\ 16 & 1 & 0 & 81 \\ 2i & 1i & 0 & 3 \end{bmatrix}$$

For obvious reasons the functions should not be called without vectorization!

$$\text{PRGM1}(Data) = \begin{bmatrix} 9.899 & 98 & \begin{bmatrix} 2i \\ 1i \\ 0 \\ 3 \end{bmatrix} \end{bmatrix}$$

$$\text{PRGM2}(Data) = \begin{bmatrix} 9.899 \\ 98 \\ \begin{bmatrix} 2i \\ 1i \\ 0 \\ 3 \end{bmatrix} \end{bmatrix}$$