Vectorization in Matlab

COS429, Fall 2013

http://www.mathworks.com/help/matlab/matlab_prog/vectorization.html

“Premature optimization is the root of all evil.”

–Donald Knuth

```
k = 0;
for t = 0:.01:10
    k = k + 1;
    y(k) = sin(t);
end
```

Elapsed time is 0.003279 seconds.

```
k = 0;
for t = 0:.01:10
    k = k + 1;
    y(k) = sin(t);
end
```

Elapsed time is 0.003279 seconds.

```
t = 0:.01:10;
y = sin(t);
```

Elapsed time is 0.000089 seconds.

```
k = 0;
for t = 0:.01:10
    k = k + 1;
    y(k) = sin(t);
end
```

Elapsed time is 0.003279 seconds.

```
t = 0:.01:10;
y = sin(t);
```

Elapsed time is 0.000089 seconds.

Always a good idea

```
x = 1:10000;
ylength = floor(length(x) / 5);
y(1:ylength) = 0;
for n = 5:5:length(x)
    y(n / 5) = sum(x(1:n));
end
```

Elapsed time is 0.029354 seconds.

```
x = 1:10000;
ylength = floor(length(x) / 5);
y(1:ylength) = 0;
for n = 5:5:length(x)
    y(n / 5) = sum(x(1:n));
end
```

Elapsed time is 0.029354 seconds.

```
x = 1:10000;
xsums = cumsum(x);
y = xsums(5:5:length(x));
```

Elapsed time is 0.000157 seconds.

```
x = 1:10000;
ylength = floor(length(x) / 5);
y(1:ylength) = 0;
for n = 5:5:length(x)
    y(n / 5) = sum(x(1:n));
end
```

Elapsed time is 0.029354 seconds.

```
x = 1:10000;
xsums = cumsum(x);
y = xsums(5:5:length(x));
```

Elapsed time is 0.000157 seconds.

"More" computation is sometimes less

```
number_to_zero_out = randi(100);
x = randi(100, 1, 10000000);
for k = 1:10000000
  if (x(k) == number_to_zero_out)
    x(k) = 0;
  end
end
```

Elapsed time is 0.041627 seconds.

```
number_to_zero_out = randi(100);
x = randi(100, 1, 10000000);
for k = 1:10000000
  if (x(k) == number_to_zero_out)
    x(k) = 0;
  end
end
```

Elapsed time is 0.041627 seconds.

```
number_to_zero_out = randi(100);
x = randi(100, 1, 10000000);
x(x == number_to_zero_out) = 0;
```

Not faster
More readable

Elapsed time is 0.049056 seconds.

```
number_to_zero_out = randi(100);
x = randi(100, 1, 10000000);
for k = 1:10000000
  if (x(k) == number_to_zero_out)
    x(k) = 0;
  end
end
```

Elapsed time is 0.041627 seconds.

```
number_to_zero_out = randi(100);
x = randi(100, 1, 10000000);
x(x == number_to_zero_out) = 0;
```

Not faster
More readable

Elapsed time is 0.049056 seconds.

Use the various Indexing Methods

```
numbers_to_zero_out = randi(100, 1, 50);
x = randi(100, 1, 10000);
for k = 1:10000
  if (ismember(x(k), numbers_to_zero_out))
    x(k) = 0;
  end
end
```

Elapsed time is 0.835425 seconds.

```
numbers_to_zero_out = randi(100, 1, 50);
x = randi(100, 1, 10000);
for k = 1:10000
  if (ismember(x(k), numbers_to_zero_out))
    x(k) = 0;
  end
end
```

Elapsed time is 0.835425 seconds.

```
numbers_to_zero_out = randi(100, 1, 50);
x = randi(100, 1, 10000);
x(ismember(x, numbers_to_zero_out)) = 0;
```

Elapsed time is 0.001615 seconds.

```
numbers_to_zero_out = randi(100, 1, 50);
x = randi(100, 1, 10000);
for k = 1:10000
  if (ismember(x(k), numbers_to_zero_out))
    x(k) = 0;
  end
end
```

Elapsed time is 0.835425 seconds.

```
numbers_to_zero_out = randi(100, 1, 50);
x = randi(100, 1, 10000);
x(ismember(x, numbers_to_zero_out)) = 0;
```

Elapsed time is 0.001615 seconds.

Functions will usually take matrix input

```
x = randi(100, 1, 10000000);
count = 0;
for k = 1:10000000
  if (x(k) == 42)
    count = count + 1;
  end
end
```

Elapsed time is 0.246667 seconds.

```
x = randi(100, 1, 10000000);
count = 0;
for k = 1:10000000
  if (x(k) == 42)
    count = count + 1;
  end
end
```

Elapsed time is 0.246667 seconds.

```
x = randi(100, 1, 10000000);
count = sum(x == 42)
```

Elapsed time is 0.218196 seconds.

```
x = randi(100, 1, 10000000);
count = 0;
for k = 1:10000000
  if (x(k) == 42)
    count = count + 1;
  end
end
```

Elapsed time is 0.246667 seconds.

```
x = randi(100, 1, 10000000);
count = sum(x == 42)
```

Elapsed time is 0.218196 seconds.

Comparisons and boolean operations can take matrix form

```
x = randi(100, 1, 10000000);
count = 0;
for k = 1:10000000
  if (x(k) == 42)
    count = count + 1;
  end
end
```

Elapsed time is 0.246667 seconds.

```
x = randi(100, 1, 10000000);
count = sum(x == 42)
```

Elapsed time is 0.218196 seconds.

Comparisons and boolean operations can take matrix form | Useful functions: all(), any()

```
x = -2:0.2:2;
y = -1.5:0.2:1.5;
for y_ind = 1:length(y)
  for x_ind = 1:length(x)
    F(y_ind, x_ind) = x(x_ind) * exp(x(x_ind) ^ 2 …
                                    - y(y_ind) ^ 2);
  end
end
```

Elapsed time is 0.000156 seconds.

```
x = -2:0.2:2;
y = -1.5:0.2:1.5;
for y_ind = 1:length(y)
  for x_ind = 1:length(x)
    F(y_ind, x_ind) = x(x_ind) * exp(x(x_ind) ^ 2 …
                                    - y(y_ind) ^ 2);
  end
end
```

Elapsed time is 0.000156 seconds.

```
x = -2:0.2:2;
y = -1.5:0.2:1.5;
[X,Y] = meshgrid(x, y);
F = X .* exp(-X .^ 2 - Y .^ 2);
```

Elapsed time is 0.000136 seconds.

```
x = -2:0.2:2;
y = -1.5:0.2:1.5;
for y_ind = 1:length(y)
  for x_ind = 1:length(x)
    F(y_ind, x_ind) = x(x_ind) * exp(x(x_ind) ^ 2 …
                                    - y(y_ind) ^ 2);
  end
end
```

Elapsed time is 0.000156 seconds.

```
x = -2:0.2:2;
y = -1.5:0.2:1.5;
[X,Y] = meshgrid(x, y);
F = X .* exp(-X .^ 2 - Y .^ 2);
```

Elapsed time is 0.000136 seconds.

Useful function: meshgrid()

```
x = randi(100, 1000, 1000);
y = randi(100, 10, 10);
for x_row = 1:1000
  for x_col = 1:1000
    y_row = mod(x_row, 10);
    if (y_row == 0) y_row = 10; end
    y_col = mod(x_col, 10);
    if (y_col == 0) y_col = 10; end
    x(x_row, x_col) = x(x_row, x_col) + y(y_row, y_col);
  end
end
```

Elapsed time is 0.199806 seconds.

```
x = randi(100, 1000, 1000);
y = randi(100, 10, 10);
for x_row = 1:1000
  for x_col = 1:1000
    y_row = mod(x_row, 10);
    if (y_row == 0) y_row = 10; end
    y_col = mod(x_col, 10);
    if (y_col == 0) y_col = 10; end
    x(x_row, x_col) = x(x_row, x_col) + y(y_row, y_col);
  end
end
```

Elapsed time is 0.199806 seconds.

```
x = randi(100, 1000, 1000);
y = randi(100, 10, 10);
y = repmat(y, 100, 100);
x = x + y;
```

Elapsed time is 0.028495 seconds.

```
x = randi(100, 1000, 1000);
y = randi(100, 10, 10);
for x_row = 1:1000
  for x_col = 1:1000
    y_row = mod(x_row, 10);
    if (y_row == 0) y_row = 10; end
    y_col = mod(x_col, 10);
    if (y_col == 0) y_col = 10; end
    x(x_row, x_col) = x(x_row, x_col) + y(y_row, y_col);
  end
end
```

Elapsed time is 0.199806 seconds.

```
x = randi(100, 1000, 1000);
y = randi(100, 10, 10);
y = repmat(y, 100, 100);
x = x + y;
```

Elapsed time is 0.028495 seconds.

Useful function: repmat()

```
x = randi(100, 1000, 1000);
sum_5s = zeros(200000, 1);
current = 1;
for row = 1:1000
  for col = 1:5:1000
    sum_5s(current) = sum(x(row, col:(col + 4)));
    current = current + 1;
  end
end
```

Elapsed time is 0.502486 seconds.

```
x = randi(100, 1000, 1000);
sum_5s = zeros(200000, 1);
current = 1;
for row = 1:1000
  for col = 1:5:1000
    sum_5s(current) = sum(x(row, col:(col + 4)));
    current = current + 1;
  end
end
```

Elapsed time is 0.502486 seconds.

```
x = randi(100, 1000, 1000);
sum_5s = sum(reshape(x, 200000, 5), 2);
```

Elapsed time is 0.021735 seconds.

```
x = randi(100, 1000, 1000);
sum_5s = zeros(200000, 1);
current = 1;
for row = 1:1000
  for col = 1:5:1000
    sum_5s(current) = sum(x(row, col:(col + 4)));
    current = current + 1;
  end
end
```

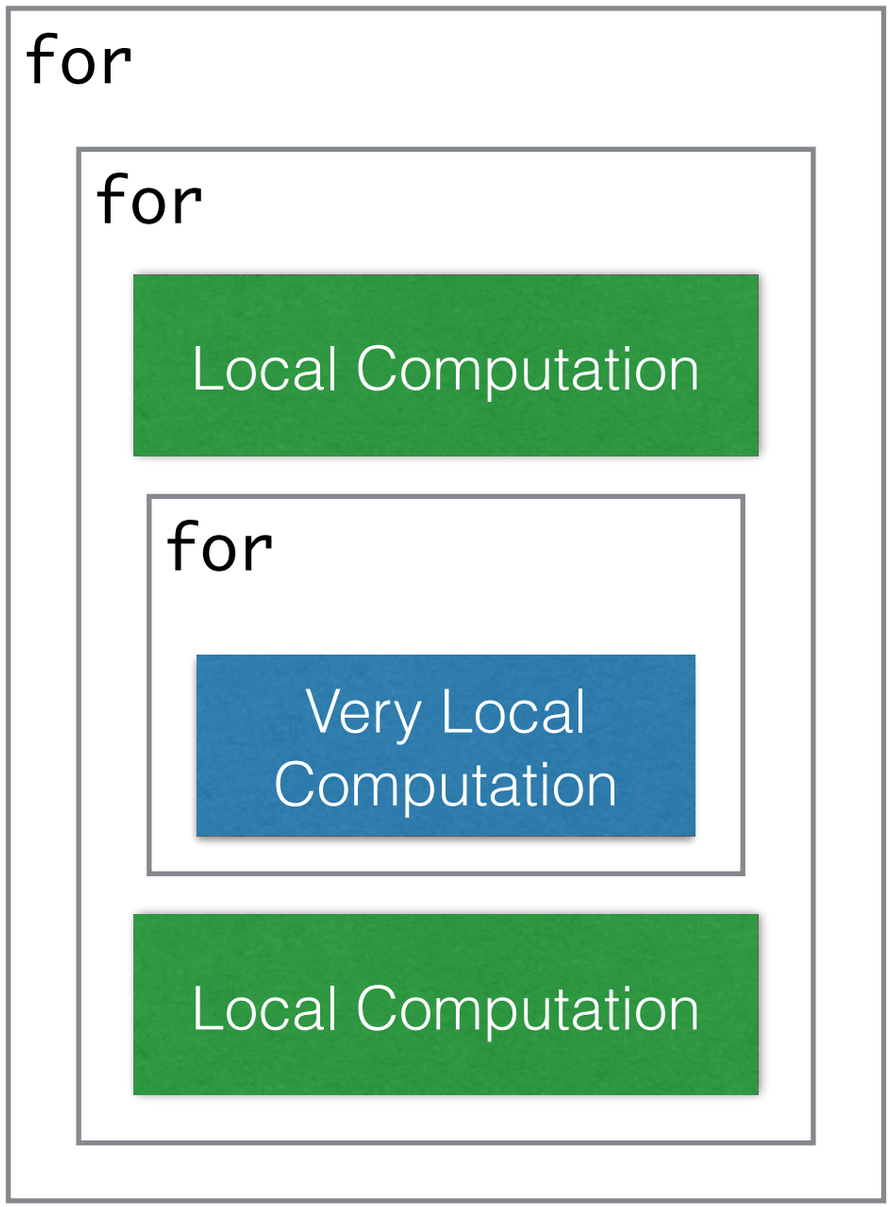
Elapsed time is 0.502486 seconds.

```
x = randi(100, 1000, 1000);
sum_5s = sum(reshape(x, 200000, 5), 2);
```

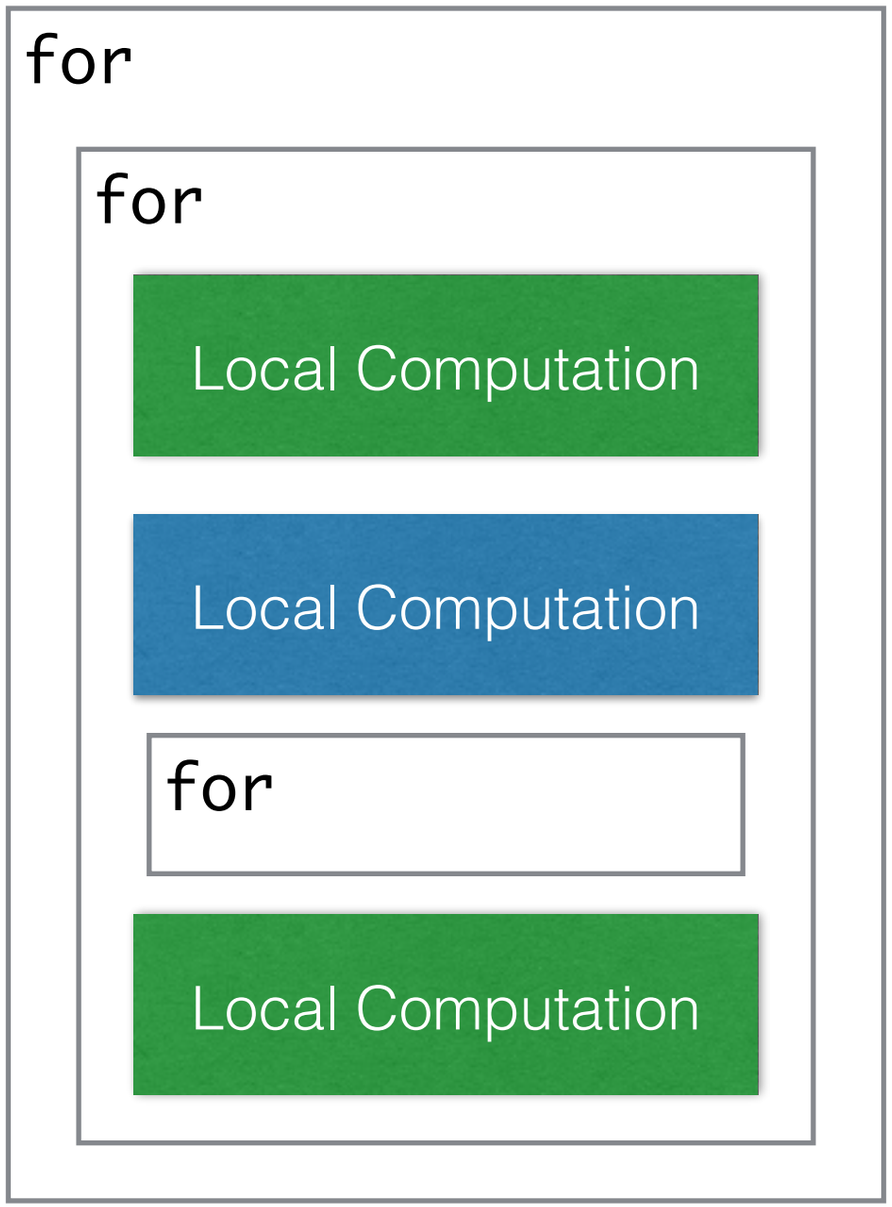
Elapsed time is 0.021735 seconds.

Useful function: reshape()

General Strategy

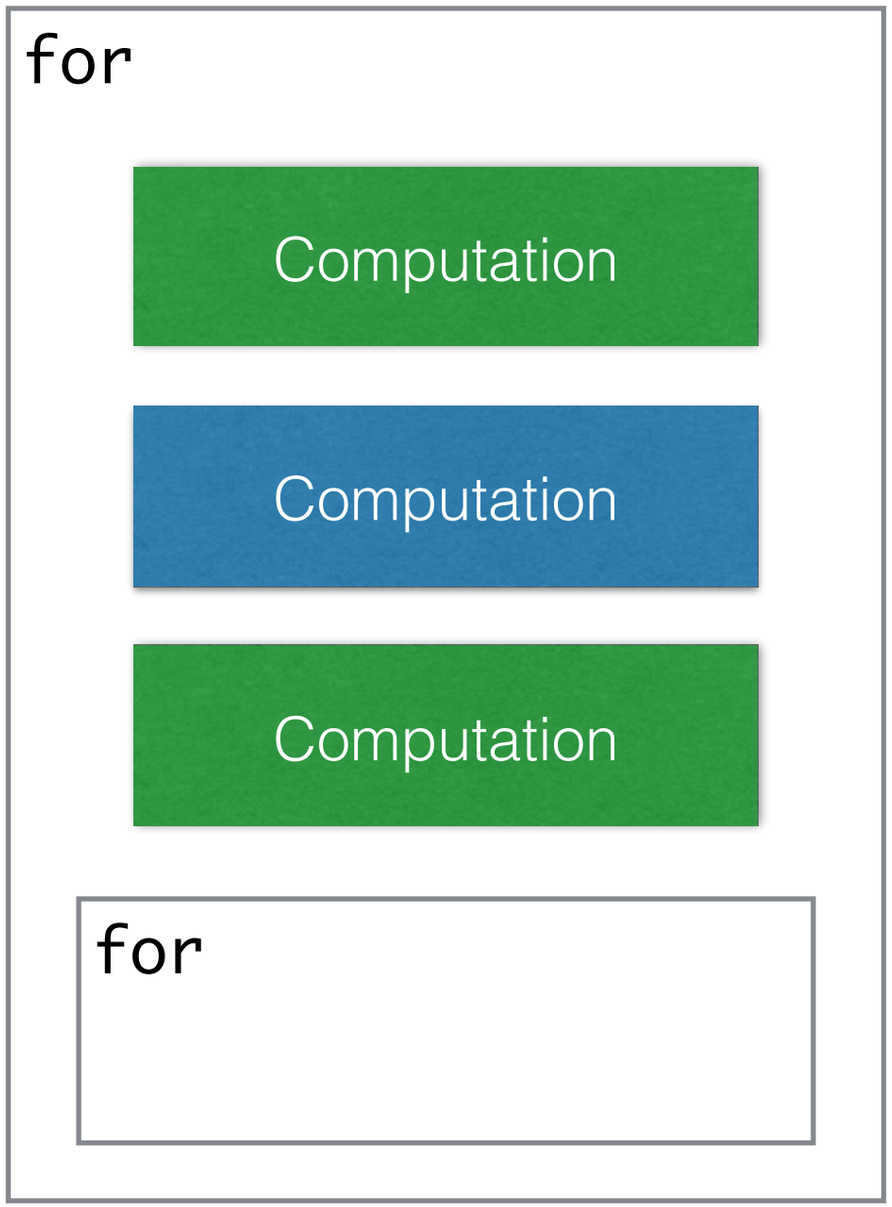


General Strategy

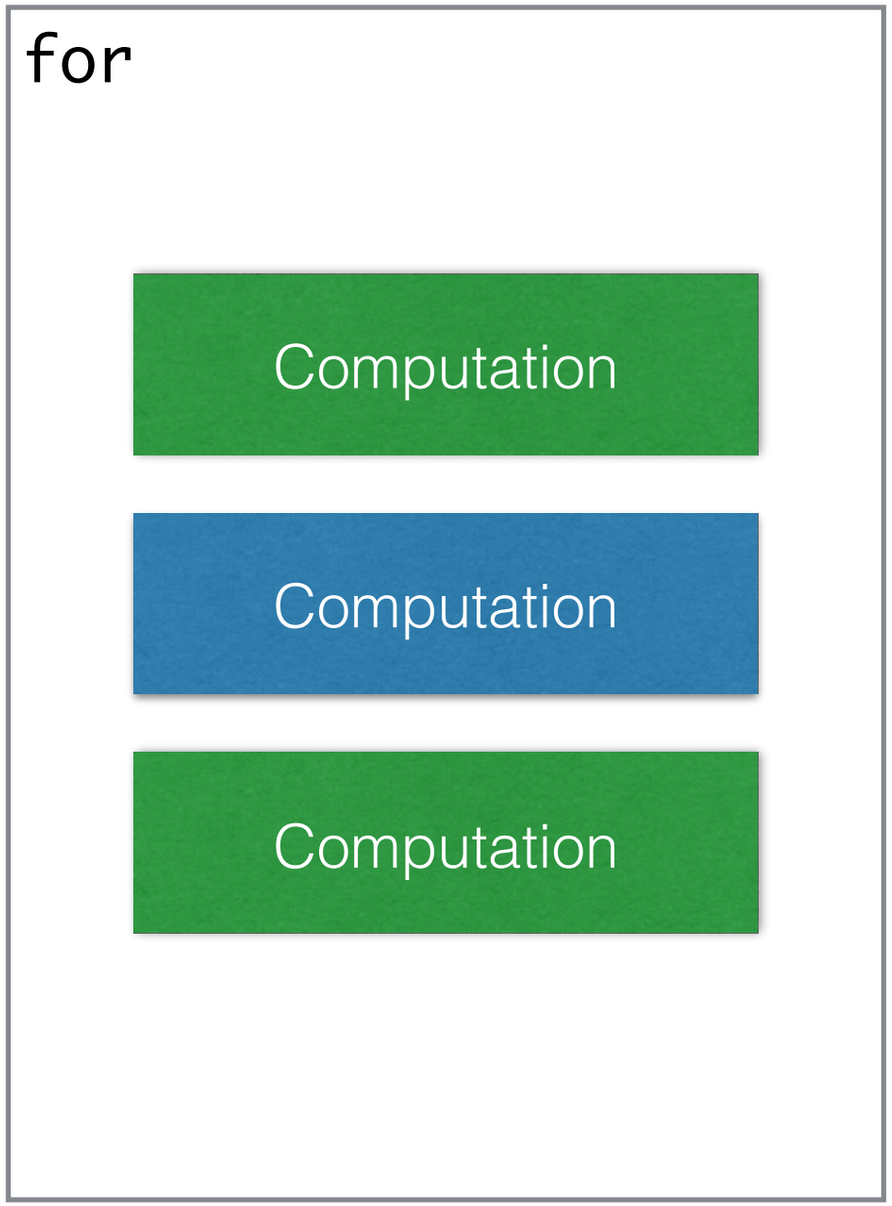


General Strategy

General Strategy



General Strategy



General Strategy

Global Computation

Global Computation

Global Computation

```
for
```

General Strategy

Global Computation

Global Computation

Global Computation