Chapter 5
Nested Loops
Part 2
sum = 0;
for j = 1 : 2
    disp( 'outer' );
    for k = 1:3
        disp( k );
        sum = sum + k;
    end
end
sum
Preallocating a Vector

- Reserves enough memory for a vector

```matlab
numCSE7 = 100;
quz3 = zeros( 1 : numCSE7 );
for k = 1 : numCSE7
    inputQz = input(`Enter quiz3 score: `);
quz3 = inputQz;
end
```

- Extending a vector (inefficient alternative)
  - Requires finding new memory
  - Copying values every time
Loops and Images

```matlab
im = imread('blacklab.jpg');
for row = [2 66 101 11 43 35 85 170]
    im( row, :, : ) = 0;
end
```

• What happened?
Nested Loops and Matrices

```
[r c] = size(matrixVar);
for row = 1 : r
    for col = 1 : c
        % Manipulate matrixVar(row, col)
    end
end
```

- Nested loop iterates through the matrix \textit{row-by-row}
- Reversing for statements iterates \textit{column-by-column}
Vectorization

- Re-writing code without loops (programming languages)
- Use MATLAB operations (every matrix element) is more efficient
  - Scalar and array operations (vec = vec + 3)
  - Logical vectors
  - Built-in functions
  - Preallocation of vectors
Vectorization of loops

- **Vectorized** version (no loops)
  - Relies on MATLAB’s fancy ways of indexing matrices/vectors

```matlab
vec = vec + 2;
```