MA1710: Key points in week 3 Matlab session

Factorials and a `break` statement

You can leave a loop before the end with a `break` statement and usually this will involve a test which has the reason for leaving the loop. An example of using `break` is as follows.

```
for n=1:30
  v=factorial(n);
  fprintf(’n=%2d, n!=%14d=%22.14e\n’, n, v, v);
  if v>=1e12
    break;
  end
end
```

Here `factorial` is a Matlab function. In this case the `break` statement is executed the first time that a factorial exceeds 10^{12}.

Using [and] to create vectors

```
a=pi

b=[5.1, 4, 3.3, -2.42, 1]

c=[5; 4; 3]

d=[1, 2, 3, 4]'
```

---

a is a variable (a 1×1 matrix).

b is a row vector.

c and d are column vectors.

Using : and using `linspace`

```
e=0:0.5:pi
```

```
e2=0:pi/6:pi
```

```
e3=linspace(0, pi, 7)
```

All generate row vectors. The last entry in `e` is 3 with `pi` just being a bound.

Using the entries

We can refer to individual entries and we can change individual entries.

```
x=0:0.2:1
```

```
x(3)
```

```
x(6)=x(6)+0.5
```

```
x(end)
```

Adding vectors, multiplying by a scalar ...

```
x=ones(1, 6)

y=2:7

z=x+y

x3=3*x

v=y-0.5
```

Evaluating a function at $x_1, \ldots, x_n$

Consider evaluating the following at points in $[0, 3]$.

$$y = x^2 - 3x + 2 = (x-2)(x-1).$$

Using a `for` loop

```
x=0:0.25:3;

m=length(x);

y=zeros(1, m);

for k=1:m
  y(k)=x(k)^2-3*x(k)+2;
end

[x; y]'
```

Evaluating a function at $x_1, \ldots, x_n$

Consider evaluating the following at points in $[0, 3]$.

$$y = x^2 - 3x + 2 = (x-2)(x-1).$$

A vectorised version

```
x=0:0.25:3;

y=x.^2-3*x+2;

[x; y]'
```

Evaluating a function at $x_1, \ldots, x_n$

Consider evaluating the following at points in $[0, 3]$.

$$y = x^2 - 3x + 2 = (x - 2)(x - 1).$$

Another vectorised version

```
x=0:0.25:3;

y=(x-2).*(x-1);

[x; y]'
```