1. One way of estimating the number $\pi$ is by computing the ratio of areas of a circle enclosed in a square and the square. The two steps below are detailed instructions of how to use MATLAB vector facilities to achieve this goal.
   a. Type `help meshgrid` to understand the command $[X \ Y] = \text{meshgrid}(x, y)$. Given row vectors $x$ and $y$, write two MATLAB statements that compute $X$ and $Y$ using only `repmat` and `length`.

   **ANSWER:**
   
   \[
   X = \text{repmat}(x, \text{length}(x), 1); \\
   Y = \text{repmat}(y', 1, \text{length}(y));
   \]

   b. Compute an approximation to $\pi$ using only MATLAB’s `meshgrid`, `linspace`, `prod`, `size`, `length`, `find`, and `sum` functions. This computation must involve no more than five MATLAB statements [the original mistakenly said four lines]. Repeat the computation to show that the approximation improves with larger meshgrids.

   **ANSWER:**
   
   \[
   n = 200; \\
   x = \text{linspace}(-1, 1, n); \\
   [X Y] = \text{meshgrid}(x, x); \\
   c = X.^2 + Y.^2 <= 1; \ % \text{solid boolean circle} \\
   \text{mypi} = 4 * \text{length}(\text{find}(c))/\text{prod}(\text{size}(c));
   \]

   Repeat with larger $n$ for better `mypi`.

2. Write a function $[d \ a \ t] = \text{interncoord}(r)$ that computes all the distances, bond angles, and torsions of a given protein chain, as defined in lecture 1. Be especially careful about the bond angle dot product $\vec{r}_{i,i-1} \cdot r_{i,i+1}$. No loops are allowed; use `diff`, `sum`, `repmat`, `cross`, `hist`, etc. [To read the C chain of 1E4K, delete the A and B chains and change ‘A’ to ‘C’ in the `if` statement halfway into `pickCA.m`]. Use `interncoord` to analyze proteins 1MBC, 155C, and 1E4K_C (the *_C means the third peptide chain in the file). Construct histogram plots for the internal coordinates for each of the proteins. Discuss similarities and differences between the proteins.

   **ANSWER:**
   
   function $[d, a, t] = \text{interncoord}(r)$
   
   % $[d, a, t] = \text{interncoord}(r)$. Compute internal coordinates of a % chain.
   %
   % r: [n 3]: protein chain % d: [n-1, 1]: bond lengths % a: [n-2, 1]: bond angles % t: [n-3, 1]: torsions
dr = diff(r);
d = sqrt(sum(dr.^2, 2));
dr = normr(dr);
dp = dot(-dr(1:end-1,:), dr(2:end,:), 2); % note minus sign
a = acos(dp);
e1 = normr(cross(dr(1:end-2,:), dr(2:end-1,:)));
e2 = normr(cross(dr(2:end-1,:), dr(3:end,:)));
t = acos(dot(e1, e2, 2));
signsin = sign(dot(cross(e1, e2, 2), dr(2:end-1,:), 2));
t(signsin < 0) = 2 * pi - t(signsin < 0);

Now we use the function:

```
r = pickCA('155C.pdb');
[d155 a155 t155] = interncoord(r);
```

```
r = pickCA('1E4K.pdb'); % multi-chain
r = r{3};
[d1e4 a1e4 t1e4] = interncoord(r);
```

```
r = pickCA('1MBC.pdb');
[d1mb a1mb t1mb] = interncoord(r);
```

% bond length histograms (normalize to get probability
distributions)
```
[n155 x] = hist(d155); % first call fixes histogram bins
n155 = n155 / length(d155); % normalize
n1e4 = hist(d1e4, x);
n1e4 = n1e4 / length(d1e4);
n1mb = hist(d1mb, x);
n1mb = n1mb / length(d1mb);
figure
plot(x, n1mb, 'r', x, n1e4, 'g', x, n155, 'b')
legend('1MBC', '1E4K_C', '155C')
hold
plot(x, n1mb, 'r.', x, n1e4, 'g.', x, n155, 'b.')
xlabel('d (Angstroms)')
ylabel('P(d)')
title('Bond length histograms (normalized)')
```

% Bond angles
```
[n155 x] = hist(a155 / pi);
n155 = n155 / length(a155);
n1e4 = hist(a1e4 / pi, x);
n1e4 = n1e4 / length(a1e4);
n1mb = hist(a1mb / pi, x);
n1mb = n1mb / length(a1mb);
figure
plot(x, n1mb, 'r', x, nle4, 'g', x, n155, 'b')
legend('1MBC', '1E4K_C', '155C')
hold
plot(x, n1mb, 'r.', x, nle4, 'g.', x, n155, 'b.')
xlabel('a (units of pi)')
ylabel('P(a)')
title('Bond angle histograms (normalized)')

% Torsion angles
[n155 x] = hist(t155 / pi);
n155 = n155 / length(t155);
nle4 = hist(tle4 / pi, x);
nle4 = nle4 / length(tle4);
n1mb = hist(t1mb / pi, x);
n1mb = n1mb / length(t1mb);
figure
plot(x, n1mb, 'r', x, nle4, 'g', x, n155, 'b')
legend('1MBC', '1E4K_C', '155C')
hold
plot(x, n1mb, 'r.', x, nle4, 'g.', x, n155, 'b.')
xlabel('t (units of pi)')
ylabel('P(t)')
title('Torsion angle histograms (normalized)')

Here’s the histogram for the bond lengths. Nothing unusual: all bond lengths are clustered strongly around the typical value of 3.8

The angle histograms are more revealing:
The most striking features are myoglobin’s (1MBC). The bond angle is strongly peaked around $\pi/2$ and the torsion around $\pi/3$. These peaks reflect myoglobin’s strongly helical character. The C chain of 1E4K, on the other hand, is dominated by beta sheets, whose bond angles and torsions cluster around $0.65\pi$ and $1.15\pi$, respectively. Values for protein 155C are clustered around both sets of peaks, which shows that it has both alpha helices and beta sheets.

3. For the above three proteins compute the contact maps. Discuss what kind of secondary elements are seen in the contact maps for the different proteins.

**ANSWER:**

```matlab
cl = dist(r1') <= 7;
figure
image(triu(cl + 1));
colormap([1 1 1; 0 0 0]);
title('Contact map for 1MBC');
```
Repeat for r2 (155C) and r3 (1E4K_C). The contact maps are below. For 1MBC we see a thick diagonal, reflecting the mainly helical character of this structure. For 1E4K, we see a slimmer diagonal (few helices, if any) and many elongated features perpendicular to the diagonal. These are beta sheets. The contact map for 155C shows that it has both kinds of secondary structure (helices and beta sheets). All three contact maps show far off-diagonal features: these are long-range contacts.