Name: __________________________________________________ (Legibly print last name, first name, middle name)

NetID: _________

Statement of integrity:
I did not, and will not, violate the rules of academic integrity on this exam.

________________________________________ (Signature)

Check here if you are in CIS121

Circle your lecture time: 9:05 or 11:15

Circle your section instructor’s name:

<table>
<thead>
<tr>
<th></th>
<th>Tuesday</th>
<th>Wednesday</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:10</td>
<td></td>
<td>Chris Scheper</td>
</tr>
<tr>
<td>11:15</td>
<td></td>
<td>Lucian Leahu</td>
</tr>
<tr>
<td>12:20</td>
<td>Lucian Leahu</td>
<td>Chris Scheper</td>
</tr>
<tr>
<td>1:25</td>
<td>Lucian Leahu</td>
<td>Chris Scheper</td>
</tr>
<tr>
<td>2:30</td>
<td>Mateo Restrepo</td>
<td>Mateo Restrepo</td>
</tr>
<tr>
<td>3:35</td>
<td>Mateo Restrepo</td>
<td>Kenneth Tsung</td>
</tr>
</tbody>
</table>

Instructions:
- This is a 90-minute, closed-book exam; no calculators are allowed.
- The exam is worth a total of 100 points, so you should try to spend no more than about 18 minutes on a 20-point question.
- Read each problem completely, including any provided code, before starting it.
- Raise your hand if you have any questions.
- Use the backs of pages or ask for additional sheets of paper as necessary.
- Clarity, conciseness, and good programming style count for credit.
- If you supply multiple answers, we will grade only one.
- Use only MATLAB code. No credit for code written in other programming languages.
- Assume there will be no input errors.
- Write user-defined functions only if asked to do so.
- Do not use cell arrays or structures.
- Do not use switch, try, catch, or break statements.
- You may find the following MATLAB predefined functions useful:
  abs, sqrt, rem, rand, floor, ceil, input, sprintf, disp

Examples:  
rem(5,2) → 1, the remainder of 5 divided by 2
rand(1, 4) → a row vector of 4 random real values, each in interval (0,1)
floor(6.9), floor(6) → 6, rounds down to the nearest integer
ceil(8.1), ceil(9) → 9, rounds up to the nearest integer
strcmp(‘At’, ‘at’) → 0, the two strings do not match; strcmp(‘at’, ‘at’) → 1
**Question 1: (30 points)**

**Part (a): (2 points)**

What does vector $v$ look like after the following script is executed?

```matlab
v = [0 1];
for k = 1:3
    v = [1 v];
end
```

*Before:* 0 1

*After:______________*

**Part (b): (2 points)**

What does vector $w$ look like after the following script is executed?

```matlab
w = [3 2 1];
w(w(3)) = w(1);
```

*Before:* 3 2 1

*After:______________*

**Part (c): (10 points)**

Assume that $a$ and $b$ are initialized scalars with $a < b$. Consider the following code fragment:

```matlab
x= linspace(a,b,n);
y= sin(x);
```

Write an equivalent fragment that does not use function `linspace` and only calls the sine function with scalar input values.

---

*Question 1 continues on next page*
Question 1, continued

Part (d): (6 points)

Assume that score is an initialized vector containing integer values in the interval [0,100]. (For example, score is a vector of student scores on a test). Write one statement on the blank below to complete the code fragment for drawing a histogram of the scores (with one bar for each score value 0, 1, 2, ..., 100).

```matlab
count = zeros(1,101);  % count will be used to store the histogram data
for k= 1:length(score)
    % statement to complete the code fragment
end
bar(0:100, count)  % draw a histogram of the scores
```

Part (e): (5 points)

Given the following function:

```matlab
function f = evaluateQuadratic(a,b,c,x)
f= a*(x^2) + b*x + c;
```

What is the output when the following script is executed?

```matlab
a=1;  b=-1;  c=3;  x=2;
f= evaluateQuadratic(c,b,a,x)
```

Part (f): (5 points)

Given the following function:

```matlab
function y = flip(x)
n= length(x);
for k= 1:n
    x(n-k+1)= x(k);
end
y= x;
```

What is the output when the following script is executed?

```matlab
y= [10 20 30 40];
y= flip(y)
```
Question 2: (20 points)

Write a function \texttt{s2hms} to convert a time in seconds to a time in hours, minutes, and seconds. The function has one parameter (\texttt{sec}) and returns three numbers: \texttt{h}, \texttt{m}, and \texttt{s}. Read the given function comment below; write the function header and the function body.

\begin{verbatim}
% Convert a time expressed in seconds (sec) to the number of hours (h),
% minutes (m), and seconds (s). h and m are integer values and
% 0\leq m,s<60. Assume sec=0.
\end{verbatim}

Assume function \texttt{s2hms} has been written correctly. Write a script to print the number of times in a day that \texttt{h>m>s}. Check whole seconds from 0 to 60\times60\times24-1. You must use function \texttt{s2hms} to solve this problem.
Question 3: (25 points)

Complete function `drawFrame` below to draw a “frame” made up of black and white disks. Each disk is of unit radius and the lower left disk is centered at (0,0). Shown on the right is an example of a 5-by-4 frame with a spacing of 0.5 between disks. The function call to produce this example is `drawFrame(5, 4, 0.5)`.

Assume that function `DrawDisk` is available. To draw a black disk of unit radius at position (3,4): `DrawDisk(3, 4, 1, 'k')`

Write only the code to draw the disks. The grid lines are provided for your convenience—you do not need to draw them.

```matlab
function drawFrame(h, w, s)
    % Draw a frame composed of h-by-w black and white disks of unit radius
    % with space s between the disks. Black disks form the border; white
    % disks are in inside. The lower left disk is centered at (0,0).
    % Assume h,w>2 and s>=0.

    axis equal
    hold on

    % Draw black disks
    for i = 1:h
        for j = 1:w
            DrawDisk(3*i, 4*j, 1, 'k')
        end
    end

    % Draw white disks
    for i = 1:h-1
        for j = 1:w-1
            DrawDisk(3*(i+1), 4*(j+1), 1, 'w')
        end
    end

    hold off
```
Question 4: (25 points)

Complete function `findPrefix(p, s)` below to return the position of the first occurrence of a word that begins with string `p` in string `s`. If no word in `s` begins with string `p`, the function returns -1. For full credit, your algorithm should be efficient—stop after the first occurrence has been found. The only built-in functions that you may use are `length` and `strcmp`. Assume that `p` contains only lower case letters and `s` contains lower case letters and blanks. Below are four examples:

<table>
<thead>
<tr>
<th>p</th>
<th>s</th>
<th>Returned value</th>
</tr>
</thead>
<tbody>
<tr>
<td>mat</td>
<td>there is a mat in the lab</td>
<td>12</td>
</tr>
<tr>
<td>mat</td>
<td>there is a bat in the lab</td>
<td>-1</td>
</tr>
<tr>
<td>mat</td>
<td>matt uses matlab on a mat</td>
<td>1</td>
</tr>
<tr>
<td>mat</td>
<td>format a plot in matlab</td>
<td>18</td>
</tr>
</tbody>
</table>

In the last example above, the word “format” in `s` includes the substring ‘mat’ but that doesn’t count since ‘mat’ does not appear in the beginning of the word.

```matlab
function k = findPrefix(p, s)
% k is the position in string s of the first occurrence of a word that
% begins with string p
% k is -1 if no word in string s begins with string p
% p contains lower case letters only
% s contains lower case letters and blanks only
```