The University of Western Ontario
Computer Science 2035b
Final Examination - Monday, April 22th, 2019
Professor: John Barron

Yes, it is true, due to class size, this exam is split over 2 rooms. Please make sure you are in
the correct room (-5% penalty is you write in the wrong room).

<table>
<thead>
<tr>
<th>HSB 35 (Health Sciences Building)</th>
<th>Akioya-J. Yan</th>
</tr>
</thead>
<tbody>
<tr>
<td>SH 3305 (Sommerville House)</td>
<td>K. Yan-Zhang</td>
</tr>
</tbody>
</table>

Please print you full name and student number, as they appear on your student card, in
the space provided below before you start this exam.

<table>
<thead>
<tr>
<th>Last Name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Given Names</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Student Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

This exam consists of 10 questions (287 pages including this page) worth a total of 360 marks
(which will be scaled to 100%). It is an open book exam, course notes and any MatLab
book(s) are allowed. No calculators, laptops or cell phones are allowed. All answers are to
be written in this booklet. Scrap work may be done on the back of each page; this will not
be marked. The exam is 180 minutes long (3 hours) and comprises 35% of your final mark.
Should your final exam grade be higher than your midterm exam grade (worth 20% of your
final grade), your final exam grade in this course will count for the full 55% of your exam grade.

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>40 marks</td>
<td></td>
<td>(6)</td>
<td>40 marks</td>
<td></td>
</tr>
<tr>
<td>(2)</td>
<td>40 marks</td>
<td></td>
<td>(7)</td>
<td>30 marks</td>
<td></td>
</tr>
<tr>
<td>(3)</td>
<td>30 marks</td>
<td></td>
<td>(8)</td>
<td>30 marks</td>
<td></td>
</tr>
<tr>
<td>(4)</td>
<td>30 marks</td>
<td></td>
<td>(9)</td>
<td>40 marks</td>
<td></td>
</tr>
<tr>
<td>(5)</td>
<td>40 marks</td>
<td></td>
<td>(10)</td>
<td>40 marks</td>
<td></td>
</tr>
<tr>
<td>Total out of 360</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
(1) (40 marks) Choose one answer (true or false) for each question. Wrong answers will be awarded -2 grade. Do not circle either true or false to receive a grade of 0.

(1) MatLab restores variables values in the workspace for a directory when MatLab is restarted in a same directory.  
true  
false

(2) The inverse of Matrices \((A*B)\) is the same as the inverse of \(B*A\).  
true  
false

(3) The dimension of the matrix produced by the MatLab statement \(\text{squeeze}(A(1:5,6,7:9))\) is 3.  
true  
false

(4) \(A=\text{zeros}(1,2,\text{'double'})\) is a 1D array.  
true  
false

(5) \(A=\text{zeros}(1,2,\text{'double'})\) is a 2D array.  
true  
false

(6) \(A=\text{zeros}(1,2,\text{'double'})\) is a column vector.  
true  
false

(7) \(A=\text{zeros}(1,2,\text{'double'})\) is a row vector.  
true  
false

(8) If \(\text{cond}(A)\) is 1558.379 then \(A\) cannot be inverted reliably.  
true  
false

(9) If \(x=1:3:9\) then \(\text{numel}(x)\) is 3.  
true  
false

(10) Is \(A=\text{zeros}(3,3,\text{'double'})\) then the size of \(A\) is 48 bytes.  
true  
false

(11) Is \(A=\text{zeros}(3,3,\text{'double'})\) then the size of \(\text{sparse}(A)\) is 72 bytes.  
true  
false

(12) Handle graphics always allows us to change the appearance of graphical entities that were plotted earlier, even if the handles of those plots are known?  
true  
false

(13) If a function \(f(x,y)\) can be symbolically integrated than it can also always be numerically integrated as well?  
true  
false

(14) If a function \(f(x,y)\) can be numerically integrated than it can also always be symbolically integrated as well?  
true  
false

(16) The MatLab statement \(x=3.0*\text{randn}(1000000,1)+3.0\) means \(x\) has a mean and standard deviation of about 3.0.  
true  
false

(17) The scientific notation of 0.0002567 is \(2.5667 \times 10^{-3}\)?  
true  
false

(18) \(\text{eps}\) is the smallest number where \(1+\text{eps}\) is zero.  
true  
false

(19) If \(Ax=B\) then \(x\) can always be computed as \(x=A/B\)?  
true  
false

(20) 123.45654 rounded to 2 digits to the right of the decimal point is 123.45?  
true  
false
(2) (40 marks) Consider a function $f(a)$ that can be computed as:

$$f(a) = \frac{\sqrt[3]{a}}{a + \sqrt[9]{a}},$$

where $a$ is a large array of floating point numbers (you have to compute the size of this array). \(\sqrt[3]{a}\) and \(\sqrt[9]{a}\) are the 3\textsuperscript{rd} and 9\textsuperscript{th} roots of $a$, which can also be computed as $a^{\frac{1}{3}}$ and $a^{\frac{1}{9}}$. Below you will be asked to write 3 MatLab functions to compute this function using the CPU and/or the GPU. Timing measurements are required using \texttt{tic} and \texttt{toc}. Only time the computation, and not any overhead required to setup the computation. Write the most efficient code you can. A run of the code below:

```matlab
function q2_2019
    % a is set here, the size of a is unknown to you.
    [f1,cp1]=calc_f1(a);
    [f2,cp2]=calc_f2(a);
    [f3,cp3]=calc_f3(a);
    fprintf('CPU sum(f1)=%12.6f cp1=%8.5f
',sum(f1),cp1);
    fprintf('VEC sum(f2)=%12.6f cp2=%8.5f
',sum(f2),cp2);
    fprintf('GPU sum(f3)=%12.6f cp3=%8.5f
',sum(f3),cp3);
end % q2_2019
```

produces the output:

```
CPU sum(f1)=83887.682627 cp1= 0.06915
VEC sum(f2)=83887.682627 cp2= 0.00970
GPU sum(f3)=83887.682627 cp3=46.25715
```

Note that the GPU solution is much slower that the CPU and VEC solutions here! This is irrelevant to the answers required here.
(2a) (10 marks) Give the MatLab code for function $[f,cp]=\text{calc}_f1(a)$ an efficient calculation of this function using arrays and loops. $f$ is the value of $f(a)$ and $cp$ is the computational measurement for this function.

Answer:

```matlab
function [f,cp]=calc_f1(a)
n=numel(a); % size(a,1) is also ok
one_third=1/3;
one_ninth=1/9;
f=zeros(size(a));

cp=tic;
for i=1:n
    f(i)=(a(i)^one_third)/(a(i)+a(i)^one_ninth);
end % i
cp=toc(cp);
end % calc_f1
```

(2b) (10 marks) Is your code in `calc_f1` standard serial code or JIT compiled code? Why or why not? An answer without justification is worth nothing. Answer:

The code will be JIT compiled because `x` is pre-allocated (`x` is computed before it is passed to `bf calc_f1`).
(2c) (10 marks) Give the MatLab code for function \([f, cp]=\text{calc}_f2(a)\), an efficient calculation of this function using vectorization. \(f\) is the value of \(f(a)\) and \(cp\) is the computational measurement for this function.

Answer:

```matlab
function [f, cp]=calc_f2(a)
one_third=1/3;
one_ninth=1/9;
end % calc_f2
```

(2d) (10 marks) Give the MatLab code for function \([f, cp]=\text{calc}_f3(a)\) an efficient calculation of the function \(\text{calc}_f1\) using the GPU. \(f\) is the value of \(f(a)\) and \(cp\) is the computational measurement for this function.

Answer:

```matlab
function [f, cp]=calc_f3(a)
n=numel(a); % size(a,1) is also ok
end % calc_f3
```
(3) (30 marks) Consider the following MatLab code:

```matlab
A=[10 5 nan 6; -5 nan 3 8; -3 inf 6 inf; 2 -7 3 1];

fprintf('Original A:
');
A
fprintf('Original sum:
');
sum(A)

A(isnan(A))=0;
fprintf('isnan A:
');
A
fprintf('isnan sum:
');
sum(A)

A(isinf(A))=0;
fprintf('isinf A:
');
A
fprintf('isinf sum:
');
sum(A)
```

What is printed by this code?

**Answer:**

```
-------------------------------
[sum works on columns not rows]
-------------------------------
```

**Original A:**

```
10 5 NaN 6
-5 NaN 3 8
-3 Inf 6 Inf
2 -7 3 1
```

**Original sum:**
<table>
<thead>
<tr>
<th>4</th>
<th>NaN</th>
<th>NaN</th>
<th>Inf</th>
</tr>
</thead>
</table>

**isnan A:**

<table>
<thead>
<tr>
<th>10</th>
<th>5</th>
<th>0</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>-5</td>
<td>0</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>-3</td>
<td>NaN</td>
<td>6</td>
<td>Inf</td>
</tr>
<tr>
<td>2</td>
<td>-7</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

**isnan sum:**

<table>
<thead>
<tr>
<th>4</th>
<th>Inf</th>
<th>12</th>
<th>Inf</th>
</tr>
</thead>
</table>

**isinf A:**

<table>
<thead>
<tr>
<th>10</th>
<th>5</th>
<th>0</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>-5</td>
<td>0</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>-3</td>
<td>0</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>-7</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

**isinf sum:**

<table>
<thead>
<tr>
<th>4</th>
<th>-2</th>
<th>12</th>
<th>15</th>
</tr>
</thead>
</table>

----------------------------------------

[but we will accept summing by rows for half marks]

----------------------------------------

**Original A:**

<table>
<thead>
<tr>
<th>10</th>
<th>5</th>
<th>NaN</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>-5</td>
<td>NaN</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>-3</td>
<td>Inf</td>
<td>6</td>
<td>Inf</td>
</tr>
<tr>
<td>2</td>
<td>-7</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

**Original sum:**

| NaN | NaN | Inf | -1  |
isnan A:

<table>
<thead>
<tr>
<th>10</th>
<th>5</th>
<th>0</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>-5</td>
<td>0</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>-3</td>
<td>Inf</td>
<td>6</td>
<td>Inf</td>
</tr>
<tr>
<td>2</td>
<td>-7</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

isnan sum:

| 21 | 6  | Inf| -1 |

isinf A:

<table>
<thead>
<tr>
<th>10</th>
<th>5</th>
<th>0</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>-5</td>
<td>0</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>-3</td>
<td>0</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>-7</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

isinf sum:

| 21 | 6  | 3  | -1 |
(4) (20 marks) This question is related to Assignment 1 in that you have to write some vectorized code (no loops) in MatLab. This question poses a calculation you did not do on the assignment. You are given the following arrays (which you may assume have been correctly computed):

1. The names cell array contains the hyphenated names of all students,

2. The exams(:,1) and exams(:,2) are the columns of the exams, giving the midterm and final exam grades as percentages (0% to 100%).

3. A lab_mark array (not on assignment 1) gives the lab marks for each student as a number from 0 to 10, while

4. An assignment_mark array (also not on the assignment) gives the assignment grades for each student as a number from 0 to 35.

Write a single vectorized MatLab line of code that prints the names of all students whose lab mark is 10 and whose assignment and midterm exam mark are at least 30 and 40 while their final mark is greater than or equal to 50 or whose assignment and final exam marks are at least 33 and 43 respectively while their final mark is greater than or equal to 53.

You may use 0 or more built-in MatLab functions in your solution. Do not use find in your solution. Use continuation dots ... to spread your statement nicely over multiple lines. Note that 26 students in the assignment 1 data satisfy this compound boolean expression.

\[
\text{names(lab_mark == 10} \& \ldots \\
\quad ((\text{assignment_mark} > 30 \& \text{exams(:,1)} > 40 \& \text{final} > 50)) | \ldots \\
\quad (\text{assignment_mark} > 33 \& \text{exams(:,2)} > 43 \& \text{final} > 53))))
\]

The output for the data on assignment 1 is:

26*1 cell array

{'Andrews-Tim' } 
{'Arif-Abubaker' } 
{'Bain-Roger' } 
{'Chen-Harold' } 
{'Chen-Yang' } 
{'Cruz-Dana' } 
{'Dai-Good' } 

Note that to logically compare vectors (which element by element), you must use & and not &&. Similarly for | and ||.
If you use && and || your get the MatLab error "Operands to the || and && operators must be convertible to logical scalar values".
(5) (40 marks) Consider the following figures. Write MatLab code segments to generate these graphs. For all graphs, print and text with fontsize 14 and boldface the labels.

(5a) (15 marks) Consider the parametric functions, \( x = \sin(2 \cdot t) \) and \( y = \sin(3 \cdot t) \). Figure 1 show a plot of these functions for \( x \in [-1 \ 1] \) and \( y \in [-5 \ 5] \).

\textbf{Answer:}

\begin{verbatim}
t=-5:0.1:5;
x = sin(2*t);
y = sin(3*t);
plot(x,t,y,t,'linewidth',2.0)
axis([-1 1 -5 5]);
% Add labels and title
xlabel('\bf \fontsize{14} x')
ylabel('\bf \fontsize{14} y')
title('\fontsize{14} x(t) = \sin(2*t), y(t) = \sin(3*t)')
\end{verbatim}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{A plot of the parametric functions.}
\end{figure}
(5b) (25 marks) Consider Figure 2 showing the Lemniscate function. Give the MatLab code to produce Figure 2 below. One version of this function gives the function as:

\[
\begin{align*}
    x &= \cos(\theta) \times \sqrt{2 \times \cos(2 \theta)} \\
    y &= \sin(\theta) \times \sqrt{2 \times \cos(2 \theta)}
\end{align*}
\]

for 100 values of \(\theta\) in the range \([-\pi/4, \pi/4]\). This draws the red part of the function shown in Figure 2 on the right. Use linewidth of 2.0, plot the right side of the function as a solid red closed curved and the left side of the function as a solid blue closed curved. Please give your answer on the next page.
Answer:

\[
\theta = \text{linspace} \left(-\pi/4, \pi/4, 100 \right) \\
\text{plot}(x, y, 'r', 'linewidth', 2.0) \\
\text{hold on} \\
\text{plot}(-x, y, 'b', 'linewidth', 2.0) \\
\text{or} \\
\text{plot}(-x, -y, 'b', 'linewidth', 2.0) \\
\text{axis('equal')} \\
\text{axis([[-1.5 1.5 -1.1 1.1]])};
\]
(6) (40 marks) Consider the following 3D plots. Draw all text with fontsize 14 and boldface all labels.

(6a) (15 marks) Consider plot of 3 parametric functions shown in Figure 3. The functions are \( x = \cos(t), y = \sin(t) \) and \( z = \sin(5t) \) for 100 equally spaced \( t \) values in the range \([-\pi, \pi]\).

Answer:

\[
\begin{align*}
\text{'\% Create the plot using the parametric functions} \\
\text{'\% } x = \cos(t), y = \sin(t), \text{ and } z = \sin(5t) \text{ for } -\pi < t < \pi \\
\text{figure} \\
t=linspace(-pi,pi,100); \\
x=cos(t); \\
y=sin(t); \\
z=sin(5*t);
\end{align*}
\]
Question (6a) continued:

```matlab
plot3(x,y,z,'-g','linewidth',2.0);
axis([%
% Add labels and title
xlabel('\bf \fontsize{14} x');
ylabel('\bf \fontsize{14} y');
zlabel('\bf \fontsize{14} z');
title('\fontsize{14} x = \cos(t), y = \sin(t), z = \sin(5*t)');
```
Question (6a) continued:
(6b) (25 marks) Consider the 3D function shown in Figure 4. The function can be specified as 3 parametric functions $x = \sin(\pi * u) * \sin(\pi * u) * \cos(v)$, $y = \sin(\pi * u) * \sin(\pi * u) * \sin(v)$ and $z = u$ with 50 values of $u$ in the range $[-1, 1]$ and 50 values of $v$ in the range $[0, 2 \pi]$. Use an azimuth angle of 135 degrees and an elevation angle of 15 degrees to display your function. [Hint: use meshgrid and vectorization in your surf calculation.] Please put your answer on the next page.

Figure 4: 3D function with equations $x = \sin(\pi * u) * \sin(\pi * u) * \cos(v)$, $y = \sin(\pi * u) * \sin(\pi * u) * \sin(v)$ and $z = u$. 
Answer:

% Create the plot using the functions
% x = sin(pi*u)*sin(pi*u)*cos(v) ; y = sin(pi*u)*sin(pi*u)*sin(v) ; z = u
% with - 1 < u < 1 and 0 < v < 2*pi
figure
u=linspace(-1,1,50);
v=linspace(0,2*pi,50);
[U,V]=meshgrid(u,v);
X=sin(pi*U).*sin(pi*U).*cos(V);
Y=sin(pi*U).*sin(pi*U).*sin(V);
Z=U;
surf(X,Y,Z);
% Add labels and title
xlabel(’\bf \fontsize{14} x’)
ylabel(’\bf \fontsize{14} y’)
zlabel(’\bf \fontsize{14} z’)
title({'’\fontsize{14} x = sin(\pi*\u)*sin(\pi*\u)*cos(\v)’, ...
’\fontsize{14} y = sin(\pi*\u)*sin(\pi*\u)*sin(\v)’, ...
’\fontsize{14} z = \u’})
% Change the view angle for the plot
% 135 degrees for azimuth angle and 15 degrees for elevation angle
view(135,15)
(7) (30 marks) This question concerns sparse matrices. Show your calculations on sparse array sizes (in bytes) for full marks. Answers without justification are worthless.

(7a) (5 marks) Consider the following MatLab code:

\[
A(5,5)=5; \\
A(10,10)=10;
\]

How many bytes are required by \(A\) as determined by `whos A`?

**Answer:**

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Bytes</th>
<th>Class</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>10x10</td>
<td>800</td>
<td>double</td>
<td></td>
</tr>
</tbody>
</table>

\[10 \times 10 \times 8 = 800 \text{ bytes}\]

(7b) (10 marks) If the MatLab code is executed to change \(A\) to a sparse matrix, i.e. \(A = \text{sparse}(A)\), how many bytes are now required by \(A\) as determined by `whos`? **Answer:**

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Bytes</th>
<th>Class</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>10x10</td>
<td>120</td>
<td>double</td>
<td>sparse</td>
</tr>
</tbody>
</table>

\[10 \times 8 \text{ columns} + (2+2) \times 8 \text{ row indices/values} + 8 \text{ nonzero number values} \]

\[80 + 32 + 8 = 120 \text{ bytes}\]

(7c) (5 marks) Consider the following MatLab code:

\[
A = \text{randn}(10,10);
\]

How many bytes are required by \(A\) as determined by `whos A`? **Answer:**

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Bytes</th>
<th>Class</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>10x10</td>
<td>800</td>
<td>double</td>
<td></td>
</tr>
</tbody>
</table>

\[10 \times 10 \times 8 = 800 \text{ bytes}\]
(7d) (10 marks) If the MatLab code is executed to change $A$ to a sparse matrix, i.e. $A = \text{sparse}(A)$, how many bytes are required by $A$ as determined by \texttt{whos}?

\textbf{Answer:}

\begin{tabular}{llll}
Name & Size & Bytes & Class & Attributes \\
A & 10x10 & 1688 & double & sparse \\
\end{tabular}

[10*8 columns + (10*10*2)*8 row indices/values + 8 nonzero number values]
[80+1600+8= 1688 bytes]

(7e) (5 marks) If we compute $A = A*A$ with $A$ being sparse, how many bytes are required by $A$ as determined by \texttt{whos}?

\textbf{Answer:}

\begin{tabular}{llll}
Name & Size & Bytes & Class & Attributes \\
A & 10x10 & 1688 & double & sparse \\
\end{tabular}

[10*8 columns + (10*10*2)*8 row indices/values + 8 nonzero number values]
[80+1600+8= 1688 bytes]
(8) (40 marks) This question concerns image processing and color edge detection and is partially based on lab 11.

Figure 5: The original Yosemite grayvalue images and its Sobel, Prewitt and Roberts edge maps.
Figure 6: The color edge map of the Yosemite image computed from the Sobel, Prewitt and Roberts edge maps.
(8a) (15 marks) Write MatLab code to compute and display the Sobel, Prewitt and Roberts edge maps using the MatLab function `edge`. Remember that `edge` returns a white edge map with black background. Display the images with a white background and black edges. Note that `edge` produces a binary image with 0 being black and 1 being white. Figure 5 above shows the original grayvalue images and the 3 edge maps. You do not have to put boxes around the edge maps as is done here.

**Answer:**

```matlab
I = imread('yos.9');
Is = edge(I, 'sobel');
Ip = edge(I, 'prewitt');
Ir = edge(I, 'roberts');

figure
imshow(I, []);
title('Yosemite Image');

figure
imshow(~Is, []);
title('Sobel Edges');

figure
imshow(~Ip, []);
title('Prewitt Edges');

figure
imshow(~Ir, []);
title('Roberts Edges');
```
(8b) (25 marks) Write MatLab code to compute a color image from these 3 edge maps. Compute a color edge map by computing red (255,0,0) from the Sobel edge map, green (0,255,0) from the Prewitt edge maps and blue (0,0,255) from the Roberts edge map. Note that black is now (0,0,0) and white is (255,255,255). It is possible for 2 of the edge maps to have a common edgel at some location, in that case, we get a secondary color. For example, if a common edgel exists in the Sobel and Prewitt edge maps then the color edgel has the yellow color (255,255,0). If all three edge maps have a common edge, then the color edgel has the color orange (255,127,0) [rather than the color white (255,255,255)]. Figure 6 shows the color edge map. There are many blue edgels but only a few red and green edgels. There are numerous yellow images (common Sobel and Prewitt edgels) but few orange edgels. **Answer:**

```matlab
Icolor=zeros(size(I,1),size(I,2),3,'double');
for i=1:size(Icolor,1)
    for j=1:size(Icolor,2)
        color=[0 0 0];
        if(Is(i,j)==1) color(1)=255; end % red
        if(Ip(i,j)==1) color(2)=255; end % green
        if(Ir(i,j)==1) color(3)=255; end % blue
        if(color(1)==255 && color(2)==255 && color(3)==255)
            color=[255 127 0];
        end
        % Change black background white background
        if(color(1)==0 && color(2)==0 && color(3)==0)
            color=[255 255 255];
        end
        Icolor(i,j,:)=color;
    end % j
end % i
figure
imshow(Icolor,[],);
title('Color Edgemap');
```
(9) (40 marks) This question is concerned least squares fitting polynomials to data. Consider the following MatLab code:

```matlab
x=linspace(0,0,7);
A=vander(x);
cond(A);
min(A(:));
max(A(:));
```

(9a) (5 marks) What is values printed by min(A(:))? Answer: 0

(9b) (5 marks) What is values printed by max(A(:))? Answer: 1

(9c) (5 marks) What is the value of matrix A? Answer:

```
0 0 0 0 0 0 1
0 0 0 0 0 0 1
0 0 0 0 0 0 1
0 0 0 0 0 0 1
0 0 0 0 0 0 1
0 0 0 0 0 0 1
0 0 0 0 0 0 1
```

(9d) (5 marks) What is the condition number of A? Answer: Inf (the matrix is singular)
(9e) (10 marks) Consider the MatLab code:

```matlab
x=0:3;
y=x.^2;
xp=linspace(0,3,4);
p1y=polyfit(x,y,3);
yorder3=polyval(p1y,xp);
figure
plot(x,y,'ok','linewidth',2.0);
figure
plot(x,y,'-ok','linewidth',2.0);
hold on
plot(xp,yorder3,'-r','linewidth',2.0);
```

Show what is plotted on the graph below (the data is already plotted as black circles). Indicate the colors of all curves that appear. **Answer:**
(9f) (10 marks) Consider the MatLab code:

```matlab
x=0:3;
y=x.^2;
xp=linspace(0,3,100);
p1y=polyfit(x,y,3);
yorder3=polyval(p1y,xp);
figure
plot(x,y,'-ok','linewidth',2.0);
hold on
plot(xp,yorder3,'-r','linewidth',2.0);
```

Show what is plotted on the graph below (the data is already plotted as black circles). Indicate the colors of all curves that appear.

**Your answer:**
(10) (35 marks) This question is based on assignment 4 of this year. Write MatLab code to draw the 2D and 3D graphs shown below. Use MatLab defaults whenever possible. Line width should be 2.0. This question tests your ability to manipulate the 4D climate array as well as writing MatLab code that draws the graphs.

(10a) (15 marks) Consider writing MatLab code to draw an errorbar graph that plots the average daily temperature for the month of February, 1965. You will need a handle for your figure. Your code should produce the graph shown below. Note that you have to write code to pull the minimum, maximum and average daily temperature data from the `climate` array.

Answer:

```matlab
% read the 4D climate array
load london_weather_1941_2013.mat 'climate';

year=1965-1940;
month=2;
LAST=climate(year,month,1,32);

min_temp=squeeze(climate(year,month,1,1:LAST));
max_temp=squeeze(climate(year,month,2,1:LAST));
ave_temp=squeeze(climate(year,month,3,1:LAST));
```
X=1:LAST;
Y=ave_temp;
U=ave_temp+max_temp;
L=ave_temp+min_temp;

Hfig=figure
errorbar(X,Y,U,L,'g-','linewidth',2.0);
hold on
plot(X,Y,'r-','linewidth',2.0);
title({'Plot of average daily temperature for February 1965';...
     'with min/max temperature errorbars'});
(10b) (15 marks) Consider plotting a 3D \textit{surf} of the average monthly temperature of London from 1941 to 2013. The graph showing this is given below. Give the MatLab code to produce this graph.

\begin{verbatim}
Answer:

% read the 4D climate array
load london_weather_1941_2013.mat 'climate';
for year=1:73
  for month=1:12
    ave_month_temp(year,month)=mean(squeeze(climate(year,month,3,1:LAST)));
  end  % month
end  % year

[XX,YY]=meshgrid(1:12,1:73);
figure
size(XX)
size(YY)
size(ave_month_temp)
surf(XX,YY,ave_month_temp);
axis([1 12 1 73 -30 20]);
title('f Average monthly temperature 1941-2013');
\end{verbatim}
(10c) (5 marks) Consider changing the graph in (10a) so that the red line is changed from red to blue through the average temperature. Use that figure's handle to do this. You should get the graph shown below.

```
Answer:
figure(Hfig)
plot(X,Y,'b-','linewidth',2.0);
```