

NAME:

MATH 451 ... EXAM 2 (IN CLASS)

1) Explain what the following commands do and draw a rough version of the figure window you would see after each command.

» subplot(2,1,1); ezplot(@(x) x.^2, [-1 1])

» subplot(2,2,3); ezplot(@(x) x.^3 - x, [-1 1])

» subplot(2,2,4); x = linspace(-2,2,20); y = x.^2 - 1 ; plot(x,y,'bo')

» subplot(1,2,1); [X Y]=meshgrid(linspace(-2,2,20);linspace(-2,2,20)); surf(X,Y,X+Y)

2) Explain what the following commands do and give the output after each command.

» $x = \text{zeros}(1,5)$

» $x = x + -3:2:6$

» $x = \text{abs}(x) + 2$

» $s1 = \text{sum}(x)$

» $p1 = \text{prod}(x)$

» $s2 = \text{sum}(\text{abs}(-3:2:6)+2)$

» $p2 = \text{prod}(\text{abs}(-3:2:6)+2)$

7
u

3) You want to create a figure with 4 plots in it. Write the commands that would plot x^2 in the upper left from $x = -2$ to 2, $x^3 - x$ in the upper right from $x = -2$ to 2, the mesh surface of $x^2 + y^2$ in the lower left from $x, y = -2$ to 2, and the contour plot of $x^2 + y^2$ in the lower right from $x, y = -2$ to 2. DO NOT use any ez-functions.

~~4) You are handed the following hand sketch of an image from a professor. Write the commands to recreate the figure.~~

5) For a vector x you can create a matrix D whose elements are the sum between the elements of x ($D = [d_{ij}]$ where $d_{ij} = x_i + x_j$). The following finds D using loops ...

```
for i = 1:length(x)
    for j = 1:length(x)
        D(i,j) = x(i) + x(j);
    end
end
```

a) Vectorize the code by removing the inner loop and include preallocation. Explain each line of your code.

b) Vectorize the code by removing both loops and include preallocation. Explain each line of your code.

6) The sequence $(1/1! + 1/3! + 1/5! + 1/7! + \dots + 1/101!)$ is evaluated using the given script ...

% s will store the value of the sequence

```
s = 1;
```

```
p = 1;
```

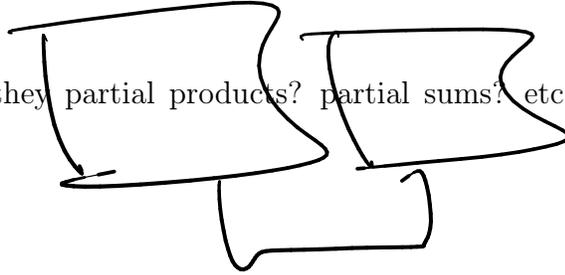
```
for i = 3:2:101
```

```
    p = p*(i-1)*i;
```

```
    s = s + 1./p;
```

```
end
```

Explain each of the lines (are they partial products? partial sums? etc) and rewrite the code using vectors and no loops.



⊛ know!

7) The following code was give to perform gaussian elimination on an augmented matrix ...

```
function [A] = badgauss(A)
n = length(A);
for i = 1:n-1
    rows = i:n;
    s = A(rows,i)/A(i,i);
    A(rows,:) = A(rows,:) - s.*A(i,:);
end
end
```

It returns the following when run ...

```
>> badgauss([1 2 3 ; 1 -2 1])
error: badgauss: A(I,J): row index out of bounds; value 3 out of bound 2
error: called from
    badgauss at line 7 column 5
```

Fix the function and explain each line of your code.

8) The following shuffle function will mix a deck of cards by removing the top 1 or 2 or 3 cards and putting them on top of the shuffled deck ...

```
function [s] = shuffle(x)
```

```
    s = [ ];
```

```
    while length(x) > 3
```

```
        n = randi(3,1);
```

```
        s = [x(1:n) s];
```

```
        x(1:n) = [ ];
```

```
    end
```

```
end
```

... is it doing what is requested? (hint: it misses some cards) Modify the function to first split the deck into a left deck and a right deck. And then shuffle by taking 1 or 2 or 3 cards from the left, 1 or 2 or 3 cards from the right, and putting them on top of the shuffled deck.