

You have 30 minutes. All paper notes are permitted, but no electronic devices of any kind may be used.

1. (5 pts) Explain the difference between `for` and `while` loops. Tell about situations in which we would prefer one over the other.

For loops iterate over the entries of a specific vector, but while loops iterate as long as a certain condition is true. We prefer for loops any time that we know beforehand how many times we need to iterate.

2. (10 pts) Give Matlab commands to create the  $87 \times 87$  matrix  $A =$
- $$\begin{pmatrix} -1 & 0 & 0 & \cdots & 0 & 0 \\ 1 & -1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & -1 & & 0 & 0 \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & 0 & 0 & \ddots & -1 & 0 \\ 0 & 0 & 0 & \cdots & 1 & -1 \end{pmatrix}.$$

```
A=-eye(87);
A(2:87,1:86)=A(2:87,1:86)+eye(86);    or    A=-eye(87);
                                           for i=1:86
                                           A(i+1,i) = 1;
                                           end
```

There are another half-dozen ways to do this.

3. (15 pts) Write Matlab commands to create a single plot of the values of all the polynomials defined by  $H_0(x) = 1$ ,  $H_1(x) = 2x$ , and  $H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x)$  for  $n = 1, 2, \dots, 11$ . Plot the polynomials at the points of the vector  $x = (-2, -1.99, -1.98, \dots, 1.99, 2)^T$ .

```
x = -2:0.01:2;
H0 = ones(1,length(x));
H1 = 2*x;
plot(x,H0,x,H1)
hold on
for n=2:11
    Hn = 2*x.*H1-2*(n-1)*H0;
    H0=H1;
    H1=Hn;
    plot(x,Hn)
end
hold off
% Could also do this by making a great big array.
```

4. (5 pts) Write Matlab commands to create the title “Hermite polynomials up to degree 11” for the current plot, where the number 11 is actually the value of a variable `deg` that gives the highest degree. In other words, if you set `deg=23`, then the title changes automatically. .

```
title(sprintf('Hermite polynomials up to degree %f',deg))  
A better format spec would be %d, but we did not discuss that in class.
```

5. (15 pts) Write a Matlab *function* called `is_zero` that takes a vector  $v$  as an argument, and returns a value of 1 if all of the entries in  $v$  have absolute value less than  $10^{-15}$ , and 0 otherwise. You would call this function as e.g. `result=is_zero(v)`.

```
function isitzero = is_zero(v)  
isitzero = 1-max(abs(v)>1e-15);  
end % function  
  
% or  
  
function isitzero = is_zero(v)  
isitzero = 1;  
for i=1:length(v)  
    if abs(v(i))>=1e-15  
        isitzero = 0;  
    end  
end  
end % function  
% Can play other games with return statement, etc.
```