

## 1. Matlab Concepts and Syntax

(A) For each of the following questions, write a single Matlab statement to accomplish what is requested.

(i) Create a row vector containing the sequence of numbers: 0 2 4 6 ... 24.

```
>> A=[0:2:24];
```

(ii) Create a column vector containing: 15.2,  $\sqrt{3}$ ,  $\sin(2.4\pi)$ ,  $\log(4.21)$ .

```
>> B=[15.2; sqrt(3); sin(2.4.*pi); log(4.21)];
```

(iii) Store your last name in a cell array called "roll" in location 12.

```
>> roll(12)={'Burdell'}; %alt #1
>> roll{12}='Burdell'; %alt #2
```

(iv) Convert Celsius temperatures in columns 5 & 7 of an array "data" to Fahrenheit.

```
>> data(:,[5 7])=data(:,[5 7]).*9./5 + 32;
```

(v) Solve for column vector,  $\mathbf{x}$ , in the matrix equation where  $\mathbf{C}$  is a square matrix,  $\mathbf{I}$  is the identity matrix and  $\mathbf{b}$  is a column vector of the same size as  $\mathbf{x}$ :  $\mathbf{C}\mathbf{x} - \mathbf{I}\mathbf{x} + \mathbf{b} = 0$

```
>> x=(C-I)\(-b);
>> x=inv(C-I)*(-b);
>> x=(C-I)^-1*(-b); %BAD FORM!
```

(B) Given  $C = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$ ,  $G = \begin{bmatrix} 4 & 3 \\ 2 & 1 \end{bmatrix}$ ,  $a = \begin{bmatrix} 10 \\ 20 \end{bmatrix}$ ,  $b = [30 \quad 40]$  what is the result of executing the following commands:

(i)  $C.*G$

```
ans =
     4     9
     4     4
```

(ii)  $G'*a$

```
ans =
    80
    50
```

(iii)  $a*b$

```
ans =
    300    400
    600    800
```

(iv)  $G./C$

```
ans =
    4.0000    1.0000
    1.0000    0.2500
```

(v)  $b*a$

```
ans =
    1100
```

## 2. Program Control

Use the most effective (readable and efficient) branching and looping code to accomplish the following:

- (A) Given a GPA between 0 – 4.0, compute the equivalent letter grade (A=4.0, B=3.0...)

```

if gpa>=3.5
    LG='A';
elseif gpa>=2.5
    LG='B';
elseif gpa>=1.5
    LG='C';
elseif gpa>=0.5
    LG='D';
else
    Lg='F';
end

```

- (B) Prompt a user to enter weights and continue computing a running summation until a negative value is entered.

```

sumwt = 0;
weight=input('Enter weight (neg to end): ');
while weight >=0
    sumwt = sumwt + weight;
    weight=input('Enter weight (neg to end): ');
end

```

- (C) Fill a vector, y, with  $y = 1 - e^{-t}$  at 0.1 sec intervals from 0 to 4.9 sec followed by the values  $y = -1 + e^{-t}$  at 0.1 sec intervals from 5 to 10 sec.

```

% What was requested
k=1;
for t=0:0.1:4.9
    y(k)=1-exp(-t);
    k=k+1;
end
for t=5.0:0.1:10
    y(k)=-1+exp(-t);
    k=k+1;
end

```

```

% What was intended...
k=1;
for t=0:0.1:4.9
    y(k)=1-exp(-t);
    k=k+1;
end
for t=5.0:0.1:10
    y(k)=-1+exp(-t+5.0);
    k=k+1;
end

```

- (D) Fill a row vector, a, of length, N, with values starting with 1 and incrementing each successive value by an additive factor r. Now use this to construct an array, A, with M rows with each row increased by a multiplicative factor, p. Assume M, N, p and r are assigned values.

```

a(1)=1;
for k=2:N
    a(k)=a(k-1) + r;
end
A=a';
for k=2:M
    A(k,:)=A(k-1,:).*p;
end

```

```

a(1)=1;
for k=2:N
    a(k)=a(k-1) + r;
end
A=a';
for k=2:M
    A(k,:)=A(k-1,:).*(1+p);
end

```

### 3. Matlab m-functions

(A) The following are short answer concept questions:

(i) Are variables defined within an m-function available for use in other scripts or m-functions? Explain.

NO. Variables defined within an m-function exist only in the function workspace and are destroyed when the function completes execution. (Exception: variables defined as global.)

(ii) Can multiple variables be returned by an m-function? Explain.

YES. WE use the form of the function definition: `[a,b] = function myfunc(x)` where a & b are returned variables.

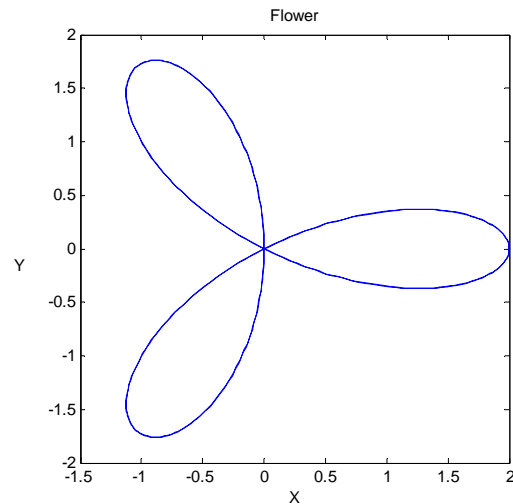
(B) A point in what is called “homogeneous space” is defined by 4 coordinates,  $(x^*,y^*,z^*,h)$ . Consider a set of N such points stored in an array with 4 columns and N rows (each point is stored in a row). In order to compute the point location in 3D space we must divide the  $x^*$ ,  $y^*$ ,  $z^*$  values of a point by its  $h$  value. In an equation:  $[x^* \ y^* \ z^* \ h] \rightarrow [x^*/h \ y^*/h \ z^*/h \ 1] = [x \ y \ z \ 1]$ . Construct an m-function called “project3D” that will consume a single  $4 \times N$  array of homogeneous points and return the same array but now for 3D space where the 4<sup>th</sup> column is all 1’s. Consider that N is arbitrary and not provided as an argument.

```
function data = project3D(data)
[r c] = size(data);
for k=1:r
    data(k,1:4)=data(k,1:4) ./ data(k,4);
end
```

#### 4. Plotting and Data Visualization

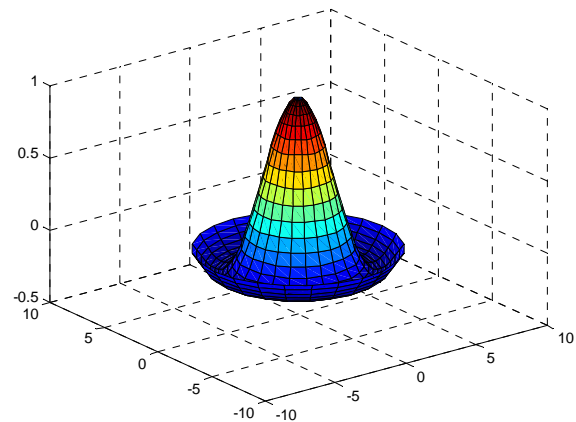
- (A) We are told that the equation:  $r = 2 \cos(3\theta)$  defines a flower when plotted for  $0 \leq \theta \leq 2\pi$ . Write a Matlab script to find out if this is true. Make sure the plot is fully labeled and titled. How many petals do you think the “flower” has?

```
theta=linspace(0,2*pi,500);
r=2.*cos(3.*theta);
x=r.*cos(theta); y=r.*sin(theta);
plot(x,y)
title('Flower')
xlabel('X'), ylabel('Y')
```



- (B) A “sombbrero” can be defined by the surface:  $z(r,\theta) = \sin(r)/r$  for  $0 \leq \theta \leq 2\pi$ . Construct a script to plot this function as a smoothly colored surface over the range:  $0 \leq r \leq 6$  and  $0 \leq \theta \leq 2\pi$  using at least 50 points in each coordinate direction. The colormap should be based on the  $z$  values. The maximum credit will be given for code that makes the best use of Matlab’s array operations. Note that  $\sin(0)/0 = 1$  but Matlab will compute NaN; you can ignore this problem (or fix it by replacing  $r=0$  with  $r=\text{eps}$ ).

```
r=linspace(0,6,25);
theta=linspace(0,2*pi,25);
[rr,tt]=meshgrid(r,theta);
xx=rr.*cos(tt);
yy=rr.*sin(tt);
zz=sin(rr)./(rr+eps); %eps optional
surf(xx,yy,zz)
```

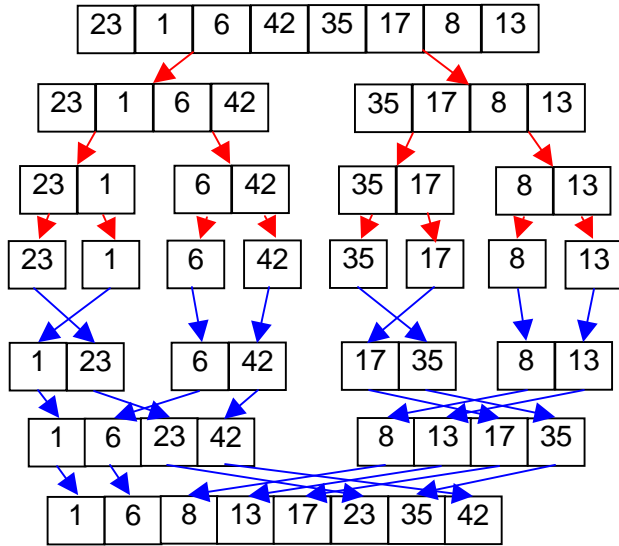


### 5. Sorting

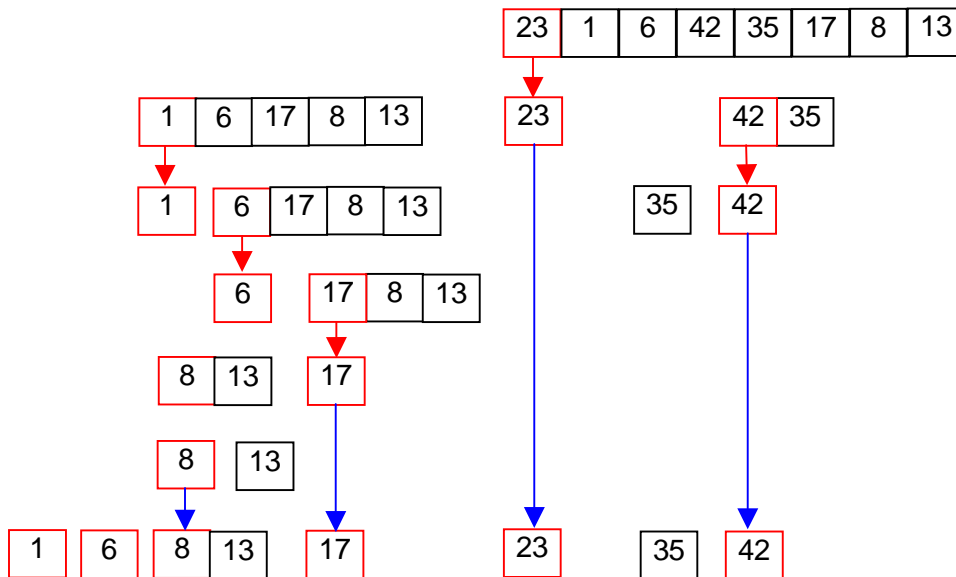
Show the steps required to sort the following set of numbers using the techniques below. You must show all the intermediate sets generated during each process.

23	1	6	42	35	17	8	13
----	---	---	----	----	----	---	----

(A) Merge Sort



(B) Quick Sort



## 6. Object Oriented Programming

(A) True or False: For each of the following statements, if the statement is true, write T. If it is false, write F, and in the space provided, write the correct information to replace the underlined word(s).

(i). F A for loop must execute the body of the code at least once.

Replacement: while

(ii). F If you pass an object to `System.out.println(...)`, its constructor is automatically called.

Replacement: toString() method

(iii). T The modifier static identifies an attribute or method belonging to the class rather than individual objects.

Replacement: \_\_\_\_\_

(iv). T It is permissible for a class to contain more than one method with the same name.

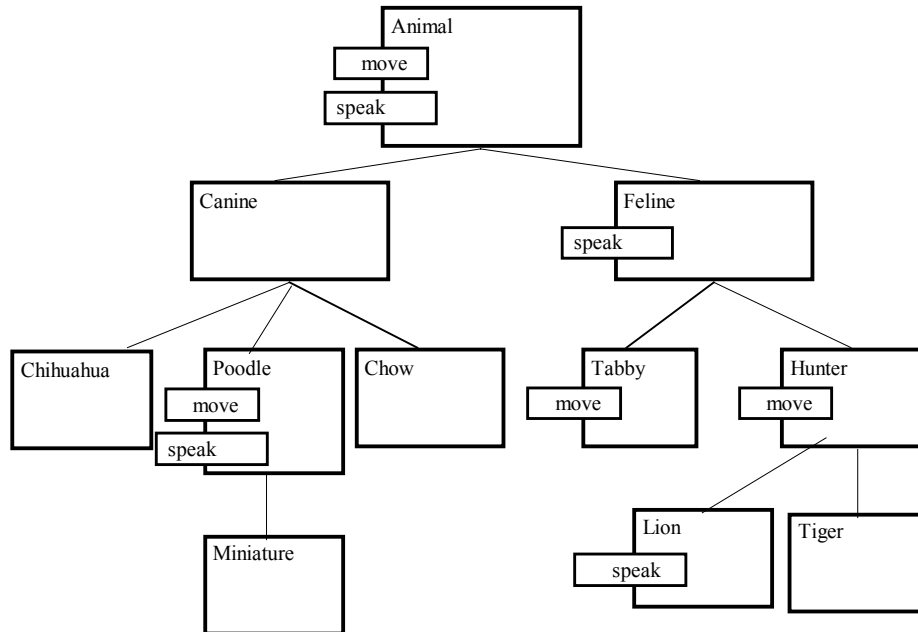
Replacement: \_\_\_\_\_

(v). F Java graphic components can be added to other components.

Replacement: containers

(continued on next page)

(B) Consider the following class hierarchy and answer the questions below:



(i) If you have a Lion object, check the “move” method would it actually use.

Animal.move()       Hunter.move()       Tabby.move()

(ii) If you have a Feline object, which “move” method would it actually use?

Animal.move()       Tabby.move()       Hunter.move()

(iii) Check the classes that are superclasses of Tiger.

Hunter       Tabby       Feline       Animal

(iv) You have the following code:

```
Animal a = new Lion();
a.speak();
```

What happens?

Animal.speak()       Lion.speak()       Java exception

(v) You have the following code:

```
Miniature p = new Canine();
p.speak();
```

What happens?

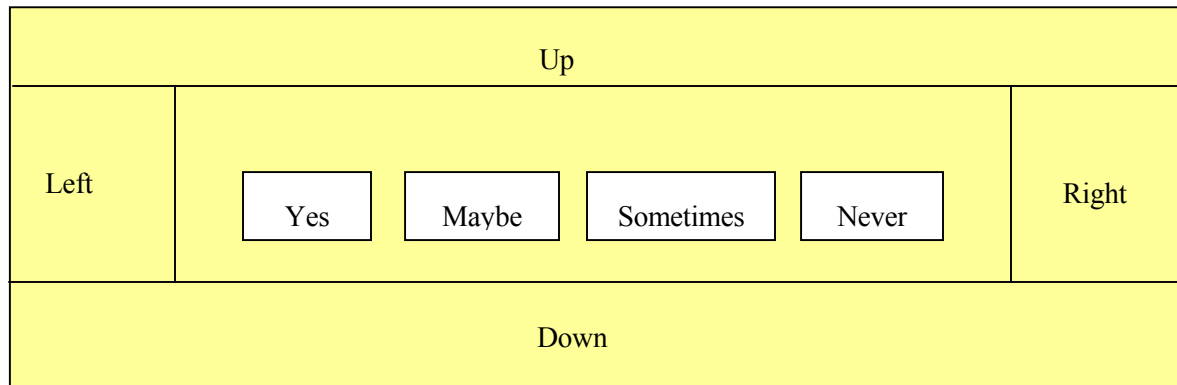
Canine.speak()       Poodle.speak()       Java exception

## 7. Java Graphics

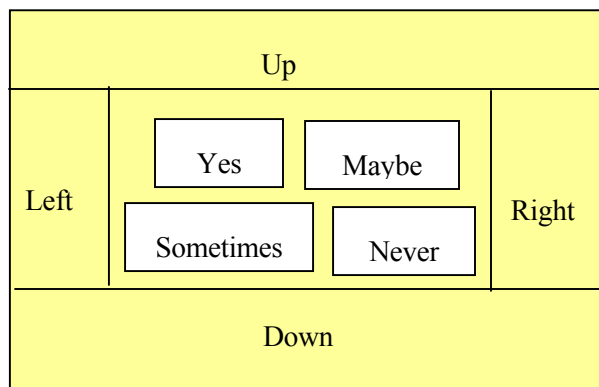
Consider the following code:

```
import java.awt.*;
import javax.swing.*;
public class GUI {
    public static void main(String[] args) {
        JFrame jf = new JFrame("GUI");
        Container c = jf.getContentPane();
        c.setLayout(new BorderLayout());
        c.add(new JButton("Up"), BorderLayout.NORTH);
        c.add(new JButton("Down"), BorderLayout.SOUTH);
        c.add(new JButton("Left"), BorderLayout.WEST);
        c.add(new JButton("Right"), BorderLayout.EAST);
        JPanel jp = new JPanel();
        jp.setLayout(new FlowLayout());
        jp.add(new JButton("Yes"));
        jp.add(new JButton("Maybe"));
        jp.add(new JButton("Sometimes"));
        jp.add(new JButton("Never"));
        c.add(jp, BorderLayout.CENTER);
        jf.setSize(600, 300);
        jf.show();
    }
}
```

(A) Sketch the picture when this code is run:

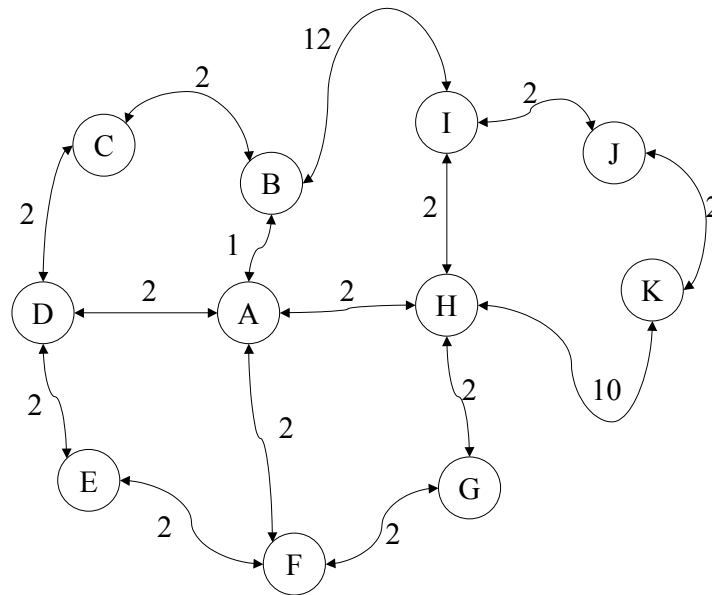


(B) Sketch the picture if you select the lower right corner with the mouse and drag it to halve the width of the window:



## 8. Searching a graph

Atlanta's streets were laid out by following indian trails through the woods, as illustrated in the following picture. We need to find the best route from A to K.



- (A) We first try a Depth-First Search (DFS), selecting the next destination in alphabetical order. What is the final path of a DFS from A to K? What is the total cost of that path?

A, B, C, D, E, F, G, H, I, J, K -> 19

- (B) Hmmmm – we suspect that might not be the best path and decide to try Breadth-First Search (BFS) instead, still using alphabetical selection. What is the final path and cost of that result?

A, H, K -> 12

- (C) We're sure that's not the best path; so you try a Greedy search (select the next destination by the least cost of travel to it). What is the final path and cost of that result?

A, B, C, D, E, F, G, H, I, J, K -> 19

- (D) What is the best path and cost of that result? What technique could possibly extract that path from this graph? Explain briefly.

A, H, I, J, K -> 8 Has to be accomplished using Dynamic Programming solving explicitly for the optimal path using Bellman's Principle of optimality to simplify the search.

## 9. Tracing

Consider the following code saved in a file called Child.java:

```
class Parent {
    int age;
    String name;
    public Parent(String n){
        name = n;
        age = 42;
        printIt("parent constructor ");
    }

    public String toString() {
        return name + " (age: " + age + ")";
    }

    public void printIt(String str){
        System.out.println(str + this);
    }
}
// -----
public class Child extends Parent{
    public Child(String n, int v){
        super(n);
        age = v;
        printIt("child constructor ");
    }

    public static void main(String[] args) {
        Child fred = new Child("Fred", 11);
        Parent sally = new Parent("Sally");
        System.out.println(fred + " is a child of " + sally);
    }
}
```

(A) Will it compile correctly using `>>! javac Child.java`?

**Yes**

(B) If not, indicate the error(s) and correct the code above.

(C) If it will compile, what will be the result if you ran the Child class using `>>! java Child`?

```
parent constructor Fred (age: 42)
child constructor Fred (age: 11)
parent constructor Sally (age: 42)
Fred (age: 11) is a child of Sally (age: 42)
```

**10. True or False**

For each of the following statements, if the statement is true, write T. If it is false, write F and in the space provided, write the correct information to replace the underlined word(s).

A. F All child classes have direct access to the parent's private data types and methods.

Replacement: public or protected

B. T Dynamic binding allows the Java Virtual Machine to choose the most appropriate method at run-time.

Replacement: \_\_\_\_\_

C. F A Stack is a FIFO data structure.

Replacement: LIFO

D. F Casting is the only way to gain access to a method in the parent class that you have redefined in the child class.

Replacement: super

E. T The constructor of a class runs only when a new object is created.

Replacement: \_\_\_\_\_

## 11. Coding

You are given the following class that contains a queue of integers. As you can see, the constructor enqueues the following set of numbers:

12, 0, 41, -3, 5, -1, 999, 17

Write the method **averageRainfall()** that will dequeue each number in turn from the queue until the 999 is retrieved. These are amounts of rainfall. Negative numbers are clearly a mistake. Your method should return the average of the positive numbers in the list. You do not need to be concerned about emptying the queue. (Hint: The average is the total of the positive numbers divided by the average of just the positive numbers.)

```
public class Rainfall {
    Queue q;

    public Rainfall() {
        q = new Queue();
    }

    public void add(int value) {
        q.enqueue(value);
    }

    public double averageRainfall() {

        double sum = 0;
        int count = 0;
        int element = q.dequeue();
        while( element != 999 ) {
            if( element >= 0 ) {
                sum += element; // or sum = sum + element;
                count++; // or count = count + 1;
            }
            element = q.dequeue();
        }
        return sum / count;
    }

    public static void main(String[] args) {
        Rainfall rnf = new Rainfall();
        rnf.add(12);
        rnf.add(0);
        rnf.add(41);
        rnf.add(-3);
        rnf.add(5);
        rnf.add(-1);
        rnf.add(999);
        rnf.add(17);
        System.out.println( "Average rainfall is: " + rnf.averageRainfall() );
    } // end of main(String[] args)
} // end of class Rainfall
```

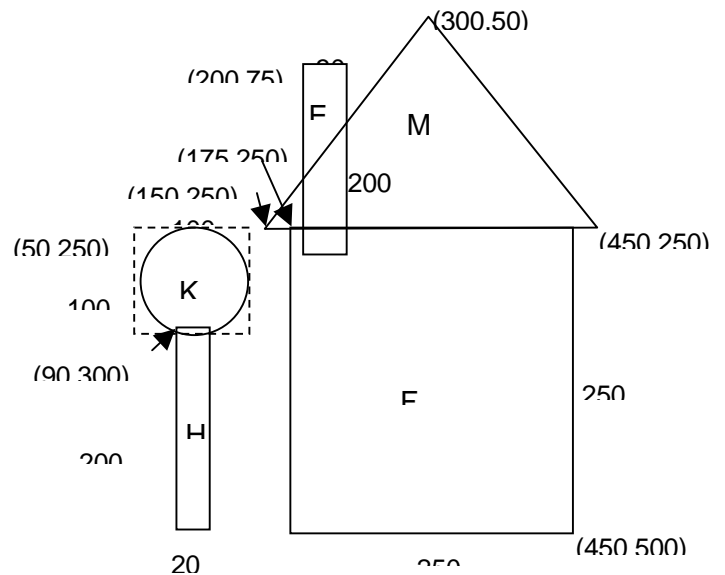
## 12. Graphics

The following code is the paintComponent() method for a canvas. In the space below, sketch the resulting picture. Label each significant location with its coordinates.

```
public void paintComponent(Graphics g) {

    Dimension d;
    int x[] = { 300, 450, 150};
    int y[] = { 50, 250, 250};

    g.setColor(Color.white);           // A
    d = getSize();                     // B
    g.fillRect(0, 0, d.width, d.height); // C
    g.setColor(Color.red);             // D
    g.fillRect(200, 75, 30, 200);      // E
    g.fillRect(175, 250, 250, 250);    // F
    g.setColor(new Color(150,50,0));    // G
    g.fillRect(90, 300, 20, 200);      // H
    g.setColor(Color.green);           // J
    g.fillOval(50, 250, 100, 100);    // K
    g.setColor(Color.black);           // L
    g.fillPolygon(x, y, 3);            // M
}
```



### 13. Matching Terms

Choose the most appropriate definition for the following terms:

Ans	Term	Definition
I	Object Oriented Programming	A. The ability of a method to elicit different behavior depending on the specific object on which it is invoked.
J	Main method	B. Getting a lot of stuff from Grannie.
M	Inheritance	C. Blueprint for behavior.
A	Polymorphism	D. Program started by Matlab.
L	FIFO	E. Selecting the most specific method for an object at run-time.
C	Class	F. Selecting the method for an object at compile-time.
E	Dynamic Binding	G. The design of a Stack
		H. Description of the methods you could invoke on an object.
		I. A programming style that encapsulates data items with the methods that manipulate them.
		J. Program started by >> !java.
		K. Simulating objects in the real world
		L. The design of a Queue
		M. The means of extending existing classes by adding methods and data.
		N. Specific instance containing data.



## 15. Matlab Concepts

Matlab defines a polynomial by storing the coefficients in a vector starting with the highest degree term and finishing with the constant term. Thus, the polynomial:  $x^2 - 3x + 2$  is stored in Matlab as  $p = [1 -3 2]$ . Write an m-function that will accept two polynomials,  $p1$  and  $p2$ , and compute their sum,  $p3$ . Your function should be used as:

```
p3 = psum(p1, p2)
```

As an example, if  $p1 = [2 1]$  and  $p2 = [1 -3 2]$ , then  $p3 = [1 -1 3]$ .

```
function p3 = psum(p1, p2)
% PSUM(p1,p2) compute sum of two polynomials

%n1=length(p1); % simple way without error check
%n2=length(p2); % simple way without error check
[r1,n1]=size(p1);
[r2,n2]=size(p2);
if (r1.*n1 ~= n1) & (r1.*n2 ~= n2)
    error('PSUM: must provide row vectors only')
end
nd = n1 - n2;
if nd > 0
    p2 = [zeros(1,nd),p2];
elseif nd < 0
    p1 = [zeros(1,nd),p1];
end
p3 = p1 + p2;
```

**16. M Functions**

Write an m-function that will compute the array product,  $C = A .* B$ . Your function should accept A and B and return C as follows: `C = amult(A, B)`. Make sure your function checks that A & B are of the same size and issue an error message if not.

```
function C = amult(A,B)
% AMULT(a,b) compute array multiplication
[rowa,cola] = size(A);
[rowb,colb] = size(B);
if (rowa ~= rowb) & (cola ~= colb)
    error('AMULT: arrays must be same size.')
end
for kr=1:rowa
    for kc=1:cola
        C(kr,kc)= A(kr,kc).*B(kr,kc);
    end
end
```

## 17. Matlab Scripts

Write a script that requests the user to enter a course average and will then compute a letter grade using IF statements. Assume that the “curve” to use is stored in a structure where: `grade.A = 90`, `grade.B = 80`, `grade.C = 70`, `grade.D = 60`, `grade.F = 0`

```
% MT1 Q3 script
avg = input('Enter course average: ');
if avg >= grade.A
    LG = 'A';
elseif avg >= grade.B
    LG = 'B';
elseif avg >= grade.C
    LG = 'C';
elseif avg >= grade.D
    LG = 'D';
elseif avg >= grade.F
    LG = 'F';
else
    LG = NaN; % optional but safer
end
disp(['Grade is ' LG]); % could use fprintf()
```

### 18. Matlab Code Analysis

The following table contains a number of different functions. You are to match up the function letter (A – G) with the description of what it does shown below.

<pre>function res = funcA(x) if x==0     res = 1; else     res = x.*funcA(x-1); end</pre>	<pre>function res = funcB(a) b=mod(a,5); if b ~= 0     res = 0; else     res = 1; end</pre>
<pre>function res = funcC(cc) A=dlmread(cc, '\t'); A(1:4, :)=[]; loc=find(A(:,5)&gt;=0); res=A(loc, :);</pre>	<pre>function res = funcD(M) [a,b]=size(M); kk=min([a,b]); res = 0; for k=1:kk     res = res + M(k,k); end</pre>
<pre>function res = funcE(raw) res(1) = {date}; res(2) = {raw(:,1)}; res(3) = {raw(:,2)}; res(7) = {raw(:,3)}; res(4) = {raw(:,4)- mean(raw(:,4))};</pre>	<pre>function res=funcF(F,x0,x1,t) y0=feval(F,x0); while abs(x1-x0) &gt; t     xmid=(x0+x1)./2;     ymid=feval(func,xmid);     if y0.*ymid &lt; 0         x1=xmid; %     else         x0=xmid;     end end res=x0;</pre>
<pre>function res = funcG(L,m1,m2,x) if m1==m2     res = m1; else;     m = fix((m1+m2)./2);     if x &lt;= L(m)         res = funcG(L,m1,m,x);     else         res = funcG(L,m+1,m2,x);     end end</pre>	

- C   Reads a tab delimited text file, **skips the first 4 rows**, and returns an array containing all data for which column #5 is greater than zero.
- D   Computes the sum of the elements along the main diagonal of an array
- A   Computes the factorial function recursively
- E   Extracts data from a cell array and returns it to the caller
- F   Finds the root of a function using the bisection method

## 19. Matlab concepts

(a) Matlab's basic data type (class) is the "array." Name as many other data types (subclasses) as you can.

Answer:

double, char, struc, cell, .are most common (also int8, uint8, int16, uint16, int32, uint32, sparse, function\_handle, user)

(b) Which Matlab data types can be used to store ANY of the data types?

Answer:

Cell and struct.

(c) Given an array created with the following statement:  $A = \text{pi} \cdot \text{rand}(8, 8)$ , write Matlab statements to:

(i) extract the 3<sup>rd</sup> column into variable x

Answer:  $x=A(:,3);$

(ii) find the sum of the diagonal elements and assign to y

Answer:  $y = \text{trace}(A)$  OR  $y=\text{sum}(\text{diag}(A))$

(iii) if the A matrix is symmetric (that is if it is equal to its transpose), assign variable sym=1, otherwise =0

Answer:  $\text{sym} = (A == A.')$

(d) Given  $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ ,  $B = \begin{bmatrix} 40 & 10 \\ 30 & 20 \end{bmatrix}$ ,  $x = [1 \ 2]$ ,  $y = \begin{Bmatrix} 3 \\ 4 \end{Bmatrix}$  what is the result of executing:

(i)  $A.*B$

Answer:  $[40 \ 20; \ 90 \ 80]$

(ii)  $A*B$

Answer:  $[100 \ 50; \ 240 \ 110]$

(iii)  $B./A$

Answer:  $[40 \ 5; \ 10 \ 5]$

(iv)  $x*y$

Answer: 11

(v)  $y*x$

Answer:  $[3 \ 6; \ 4 \ 8]$

(e) What is the difference between  $\text{eye}(4, 4) \setminus A$  and  $A \setminus \text{eye}(4, 4)$  where A is a nonsingular 4x4 matrix?

Answer: first results in A (left-divide by unity); second produces inverse of A

(f) You are given a vector (row). Construct Matlab code to shift all elements left by N positions. (ex: for N=3,  $[1 \ 2 \ 3 \ 4 \ 5 \ 6] \gg [4 \ 5 \ 6 \ 1 \ 2 \ 3]$ ).

Answer:

$A=[A(N+1:end), A(1:N)]$

**20. Matlab programs**

- (a) The following FOR loop computes  $y(k)$ . Rewrite it to avoid using the FOR loop and instead use Matlab's vector operators.
- ```
for k = 1:61
    t = 6.*pi.*(k-1)./60;
    y(k) = 5.*sin(t) - 3.*cos(2.*t);
end
```

Answer:

```
t = 0:pi./10:6.*pi;
y=5.*sin(t) - 3.*cos(2.*t);
```

- (b) Consider this algorithm: Given  $N$  (positive integer), if  $N$  is even, divide by 2, otherwise multiply by 3 and add 1 to compute a new value of  $N$ . Repeat this until the answer is 1. Write a Matlab program to compute this and to display at the end of the program: (i) the total number of iterations required, and (ii) the successive values of  $N$  in a row vector.

Answer:

```
count = 0;
result=N;
while N > 1
    if rem(N,2) == 0
        N = N/2;
    else
        N=3*N+1;
    end
    count=count+1;
    result=[result, N];
end
count, result
```

## 21. Matlab functions

(a) Describe the differences in namespaces (workspaces) for a script and a function in Matlab.

Answer:

Each Matlab function is executed in its own namespace (workspace) which is destroyed when function execution completes. All scripts execute using the namespace (in the workspace) of the Command window and share variables.

(b) Write a Matlab function, `polyadd(p1, p2)` that will add together two polynomials defined using Matlab's polynomial representation (i.e.,  $x^2 + 3x + 2 \rightarrow [1 \ 3 \ 2]$ ). You must assume the polynomials can be of different degrees and that not all terms may be present (nonzero). (i.e., if `p1=[1 3 2]`, `p2=[2 1]`, then `p=polyadd(p1,p2)` yields `p=[1 5 3]`.)

Answer:

```
function p=polyadd(p1, p2)
% Add together two polynomials
n1=length(p1);
n2=length(p2);
diff=n1-n2;
if diff > 0
    p2=[zeros(1,diff) p2];
else
    p1=[zeros(1,-diff) p1];
end
p=p1+p2;
```

**22. Plotting and data visualization**

- (a) Write a Matlab script to plot the three functions:  $f_1(x) = 0.2x^2$ ,  $f_2(x) = \sin^2(x)$ ,  $f_3(x) = e^{-0.2x}$  for  $0 \leq x \leq 3.5$  using at least 50 points. Make each curve a different color and create axis labels and a legend.

```
Answer:
x=linspace(0,3.5,50);
f1=0.2.*x.^2;
f2=(sin(x)).^2;
f3=exp(-0.2.*x);
F=[f1;f2;f3];
plot(x,F)
% ALT: plot(x,f1,x,f2,x,f3)
% ALT: plot(x,f1,'r',x,f2,'b',x,f3,'g')
xlabel('x')
ylabel('y')
legend('f1','f2','f3')
```

- (b) Construct a Matlab script to plot the surface defined by  $z = c \sin(2\pi axy)$  where  $a=3$ ,  $c=0.25$  over  $-1 \leq x, y \leq +1$ . Explain what you must do (or show code) to create a surface with smooth (interpolated) coloring.

```
Answer:
a=3; c=0.25;
x=linspace(-1,1,30);
y=x;
[xx,yy]=meshgrid(x,y);
z=c.*sin(2.*pi.*a.*xx.*yy);
surf(xx,yy,z)
shading('interp')
```

### 23. Structures and cell arrays

- (a) Write Matlab statements that will create the 5<sup>th</sup> element in a structure array, `student(5)`, containing:
- ```
name = George Burdell
SID = 123456789
prism = gta0000a
class = sophomore
term = 200301
```

Answer:

```
>> student(5).name='George Burdell';
>> student(5).SID=123456789;
>> student(5).prism='gta0000a';
>> student(5).class='sophomore';
>> student(5).term=200301;
```

- (b) Now add an additional nested structure called “classes” where for the above student you should store for `classes(3)`:
- ```
name = COE1361
section = C
instructor = Craig
exams = [72, 91, 88]
final = 93
```

Answer:

```
>> student(5).classes(3).name='COE1361';
>> student(5).classes(3).section='C';
>> student(5).classes(3).instructor='Craig';
>> student(5).classes(3).exams=[72,91,88];
>> student(5).classes(3).final=93;
```

- (c) Write Matlab code to compute the course grade for George using:  $grade=0.4*final + 0.6*(midterm\ average)$  and save this in a new field in `classes(3)` called “grade.”

Answer:

```
>> grade=0.4.*student(5).classes(3).final + 0.6.*mean(student(5).classes(3).exams)
>> student(5).classes(3).grade=grade
```

- (d) We could actually use a cell array to implement a simple stack in Matlab because a cell array can contain any type of data and arrays can easily be extended dynamically. Consider the m-function, `push(s,data)` that adds scalar numeric data to a stack, `s`, (assume `s` has already been created with another “constructor” function) and returns the new stack, `s`.

```
function s = push(s,data)
s = [s;data];
```

Modify this so `s` is a cell array and the data can then be any valid Matlab data type (arrays, strings, struct's, other cell arrays, etc). You can assume also that the stack, `s`, has already been created by a “constructor” function.

Answer:

```
function s=push(data,s)
s={s, data};
```

## 24. OO questions

a) What is an "object-oriented" programming language?

Answer:

A language that facilitates behavioral abstraction – encapsulating object-specific information with the methods to manipulate that information.

b) What is the difference between a class and an object?

Answer:

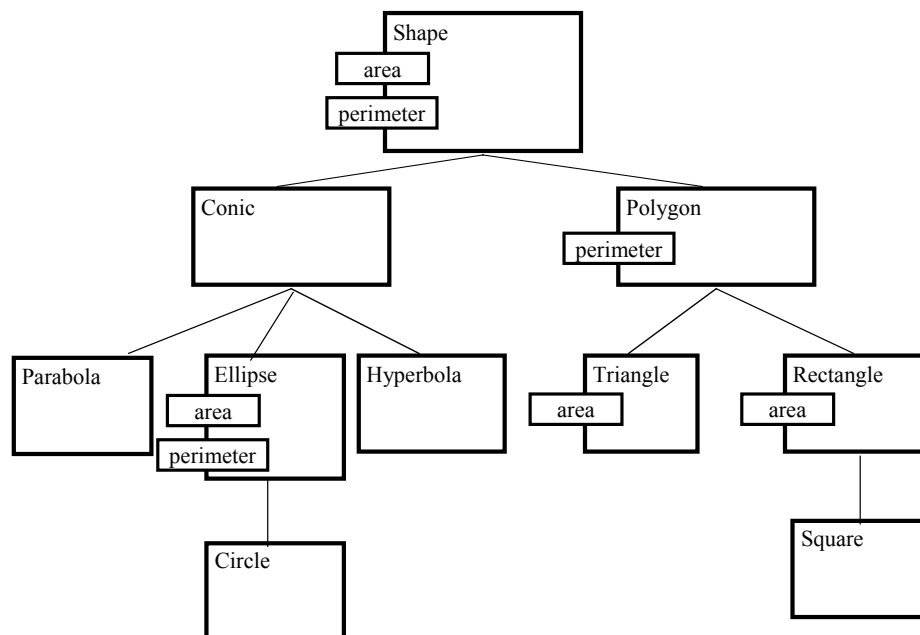
A class is like a blue print specifying the properties and behavior of real-life objects. An object is a specific instance of a class populated with specific data.

c) Is it legal to have two methods with the same name in the same class? Why?

Answer:

Yes – most OO programming languages distinguish methods not only by their name, but also by the type of data they return and by the number and type of the parameters.

d) Consider the following class hierarchy:



(i) If you have a Circle object, which “area” method would it actually use?

Answer: `Ellipse.area()`

(ii) If you have a Square object, which “perimeter” method would it actually use?

Answer: `Polygon.perimeter()`

(iii) From the diagram, name the classes that are superclasses of Triangle.

Answer: Polygon and Shape

(iv) If you asked for the area of a Parabola object, which method would be used? Why?

Answer: the default `Shape.area()` because you really can't define the area of a parabola.

e) explain the following terms:

(i) static:

Answer: the modifier on an attribute or method of a class indicating that it belongs to the class itself and not to individual objects of that class.

(ii) constructor:

Answer: the code that runs as soon as an object is created in order to initialize its attributes.

(iii) accessor:

Answer: a method provided to give public access to the value of private data items.

(iii) modifier:

Answer: either a) a method provided to allow public replacement of the value of private data items, or b) words such as 'public', 'static' or 'final' that modify users' access to methods and data.

f) briefly discuss the concept of Polymorphism in the context of a collection of Shapes.

Answer: Polymorphism allows a program to treat any collection of objects inheriting from a common parent in a uniform way. For example, if we were given a cell array of different shapes, we could iterate across that array requesting the area or perimeter of each shape without caring what specific shape it was.

## 25. Java Graphics

a) Name the three basic component types that make up a Java Swing graphics application. State their purpose and give an example of each.

1. Component: **Answer: Container**

Purpose: **Answer: a receptacle for graphic components and other containers**

Example: **Answer: JFrame, JPanel**

2. Component: **Answer: Component**

Purpose: **Answer: An actual graphic widget that can be drawn**

Example: **Answer: Canvas, Button, List, ScrollBar, TextBox, CheckBox**

3. Component: **Answer: Layout**

Purpose: **Answer: Specify the distribution of Components in a Container**

Example: **Answer: Border, Grid, Flow or Box**

b) Consider the following code fragment:

```
public void paintComponent(Graphics g) {  
    g.setColor(Color.green);  
    g.fillPolygon({ 80, 220, 200, 100 },  
                 { 100, 100, 120, 120 }, 4);  
    g.setColor(Color.black);  
    g.drawLine( 150, 10, 150, 100 );  
    g.drawLine( 100, 100, 150, 20 );  
    g.drawLine( 150, 20, 200, 100 );  
}
```

**Hint:** `fillPolygon(...)` consumes three parameters: an array of x values, an array of y values and the number of points in the polygon. It automatically closes the polygon if the first and last points are not the same.

(i) when would such a module normally run?

**Answer: each time the containing Component needs to be redrawn**

(ii) sketch the picture being drawn.

**Answer: a sail boat**

## 26. Java Code

Consider the following code:

```
public class SCoord {
    public double x;
    public double y;

    public SCoord( SCoord p ) {
        x = p.x;
        y = p.y;
    }
    // ----- A -----
    public SCoord( double px, double py ) {
        x = px;
        y = py;
    }

    public SCoord sum(SCoord c) {
        return new SCoord( x + c.x, y + c.y );
    }

    public void shift(SCoord c) {
        x = x + c.x;
        y = y + c.y;
    }

    public double distance(SCoord c) {
        double dx = x - c.x;
        double dy = y - c.y;

        return Math.sqrt((dx*dx) + (dy*dy));
    }

    public String toString() {
        return "[ " + x + ", " + y + " ]";
    }

    public static void main(String args[]) {
        SCoord a = new SCoord( 1, 2 );
        SCoord b = new SCoord( 3, 4 );
        SCoord c = a.sum(b);
        System.out.println("distance from " + a + " to " + c
            + " is " + a.distance(c) );
        a.shift(b);
        System.out.println("a is now " + a + " and b is now " + b );
    }
}
```

a) is the code module following the marker // ----- A ----- legal? Explain.

Answer:

Yes, it is legal for a class to have methods or constructors of the same name as long as the number or types of the parameters are different.

b) what would be displayed in the Matlab command window if you entered the following:

```
> !java SCoord
```

Answer:

```
distance from [ 1.0, 2.0 ] to [ 4.0, 6.0 ] is 5.0
a is now [ 4.0, 6.0 ] and b is now [ 3.0, 4.0 ]
```

c) we need to add to the SCoord class a method called *scale* that consumes a number (typed double) produces a new SCoord object with both x and y multiplied by that number. Write the method in the space below, and write a large ← B on the listing above indicating where it should be inserted.

Answer:

```
public SCoord scale(double v) {
    return new SCoord( x * v, y * v );
}
```

Place the B anywhere within the class, but outside any other methods.

d) we need to test the new *scale* method by adding code in the main module. Write in the space below the line necessary to test it by printing out the result of scaling the SCoord a by a factor of 3. Write a large ← C on the listing above indicating where this line should be inserted.

Answer:

```
System.out.println("a scaled by 3 is " + a.scale(3) );
anywhere in main(...)
```

e) write below the resulting output when you again enter this in the command window:

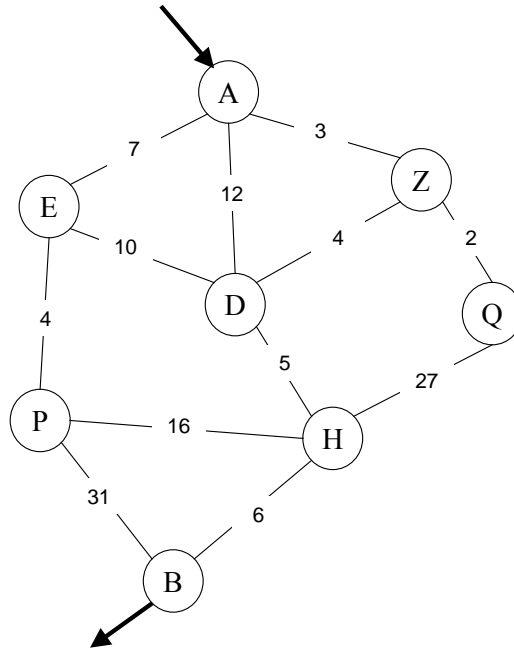
```
> !java SCoord
```

Answer:

```
distance from [ 1.0, 2.0 ] to [ 4.0, 6.0 ] is 5.0
a is now [ 4.0, 6.0 ] and b is now [ 3.0, 4.0 ]
a scaled by 3 is [ 12.0, 18.0 ]
```

## 27. Searching a graph

It's a well known fact that pirates love treasure. Captain Jim has asked you to draw up a course from the following undirected graph of interconnected islands. He wants to go from island A to island B.



a) He first suggests a Breadth-First Search (BFS), selecting adjacent islands in alphabetical order. What is the final path of a BFS from A to B? What is the total cost of that path?

Answer:

A, D, H, B -> 23

b) Hmm – he suspects that might not be the best path and suggests you try Depth-First Search (DFS) instead, still using alphabetical selection. What is the final path and cost of that result?

Answer:

A, D, E, P, B -> 57

c) Look out – he's sure that's not the best path; being pirates, you naturally suggest a Greedy search (select the next island by the least cost of travel to it). What is the final path and cost of that result?

Answer:

A, Z, Q, H, D, E, P, B -> 82

d) "Avast, ye swabs!" he yells, (pirate for "No, you idiot!") "I can see the best path with me own eyes!" What is the best path and cost of that result? What technique could possibly extract that path from this graph? Explain briefly.

Answer:

A, Z, D, H, B -> 18 Has to be accomplished using Dynamic Programming solving explicitly for the optimal path using Bellman's Principle of optimality to simplify the search.

## 28. Short answers

a). Compare the function of the Matlab “save” command and the “dmlwrite” function.

Answer:

**save** will write specified data from the Matlab base workspace to a binary file that can only be reloaded into the Matlab workspace using **load**; **dmlwrite** will write specified variable values into a delimited text file that is human readable.

b) Give an example of polymorphism in Matlab.

Answer:

The Matlab math trig functions will work with either scalar or array arguments; Matlab functions can also accept variable numbers of arguments.

c) (i) What is the file extension of Matlab data files? Answer: `.mat`

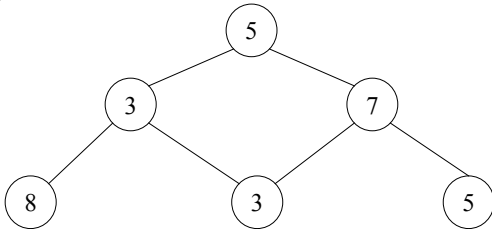
(ii) What is the file extension of Matlab script files? Answer: `.m`

(iii) What is the file extension for a file containing a compiled Java file? Answer: `.class`

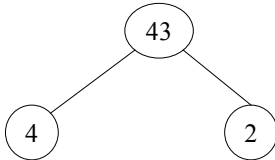
(iv) What is the file extension for a file containing java source code? Answer: `.java`

d) Which of the following is not a binary tree and why?

i)



ii)



iii)



Answer: (i) - it has a cycle

**29. Short Answers**

- a. Why is Java called an "object-oriented" programming language?

Encapsulates data and methods

- b. In Java terminology, what is the difference between a class and an object?

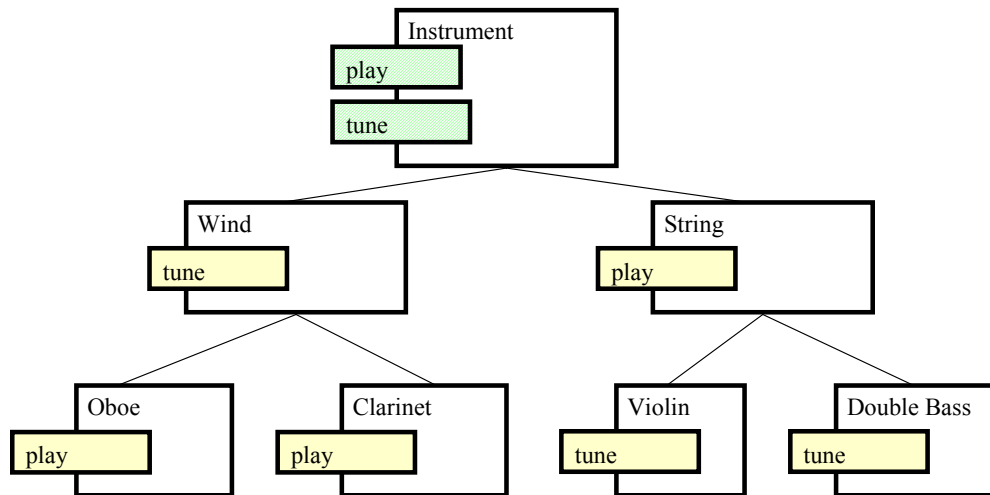
A class is the blueprint; an object is an instance of a class.

- c. In Java, is it legal to have two methods with the same name in the same class? Why?

Yes, because Java uses the name, return type and parameter types as a method signature to differentiate between methods with same name.

### 30. Inheritance

Consider the following class hierarchy:



a. If you have an Oboe object, which “**play**” method would it actually use?

**Oboe.play**

b. If you have a Violin object, which “**play**” method would it actually use?

**String.play**

c. From the diagram, name a class that is a superclass of String.

**Instrument**

### 31. Code

Consider the following code and answer the questions (a through f), each of which pertains to the code immediately following the question

```
public class SearchStack {

    private ListNode head;
    private int ID;
    // a. what does static mean? belongs to the class; not the object
    private static int numberOfStacks = 0;

    // b. What is this thing called? When does it run? Constructor; runs when you
    // make a new object.
    public SearchStack() {
        head = null;
        ID = ++numberOfStacks;
    }

    public void push(Object o) { ... } // left out the guts
    public boolean isEmpty() { return (head == null); }

    public Object pop() throws Exception {
        if(isEmpty())
            // c. What is happening here? Java complaining about a user error
            throw new Exception("Stack pop underflow");
        Object res = head.data;
        head = head.next;
        return res;
    };

    // d. what is this method for? displaying the contents of an object
    public String toString() {
        String res = "SearchStack(" + ID + ")\n";
        ListNode here = head;

        while ( here != null ) {
            // e. Why does this line of code work? because Java automatically invokes the
            // toString method on any non-string object.
            res = res + here.data + "\n";
            here = here.next;
        }
        return res;
    }

    public static void main(String[] args) {
        // f. what is the function of new? fetch/allocate memory for an object
        SearchStack s = new SearchStack();

        System.out.println("Empty stack is " + s);
    }
}
```

```
s.push("Fred");
System.out.println("With Fred on it: " + s);
s.push("Lucy");
System.out.println("With Fred and Lucy on it: " + s);
}
}
```

*g. In your MatLab command window, what happens when you type:*

```
>> my_S_queue = SearchStack()
```

*Error: parentheses are not necessary*

*h. What should you have typed?*

*my\_S\_queue = SearchStack*

*i. Why is this different from typing the following?*

```
>> !java SearchStack
```

*This runs the main method; #h just creates an object*

*l. What will you see in the interactions window when you type*

```
>> !java SearchStack
```

*Empty stack is SearchStack(1)*

*With Fred on it SearchStack(1)*

*Fred*

*With Fred and Lucy on it SearchStack(1)*

*Fred*

*Lucy*

**32. Extra Credit**

If you're finished early and bored, write a new method for the SearchStack class that will look and see if a specific object is in the stack. It will consume an Object and return a boolean result.

```
public Boolean find(Object o) {  
    ListNode here = head;  
    while (here != null) {  
        if (here.data == o) return true;  
        here = here.next;  
    }  
    return false  
}
```