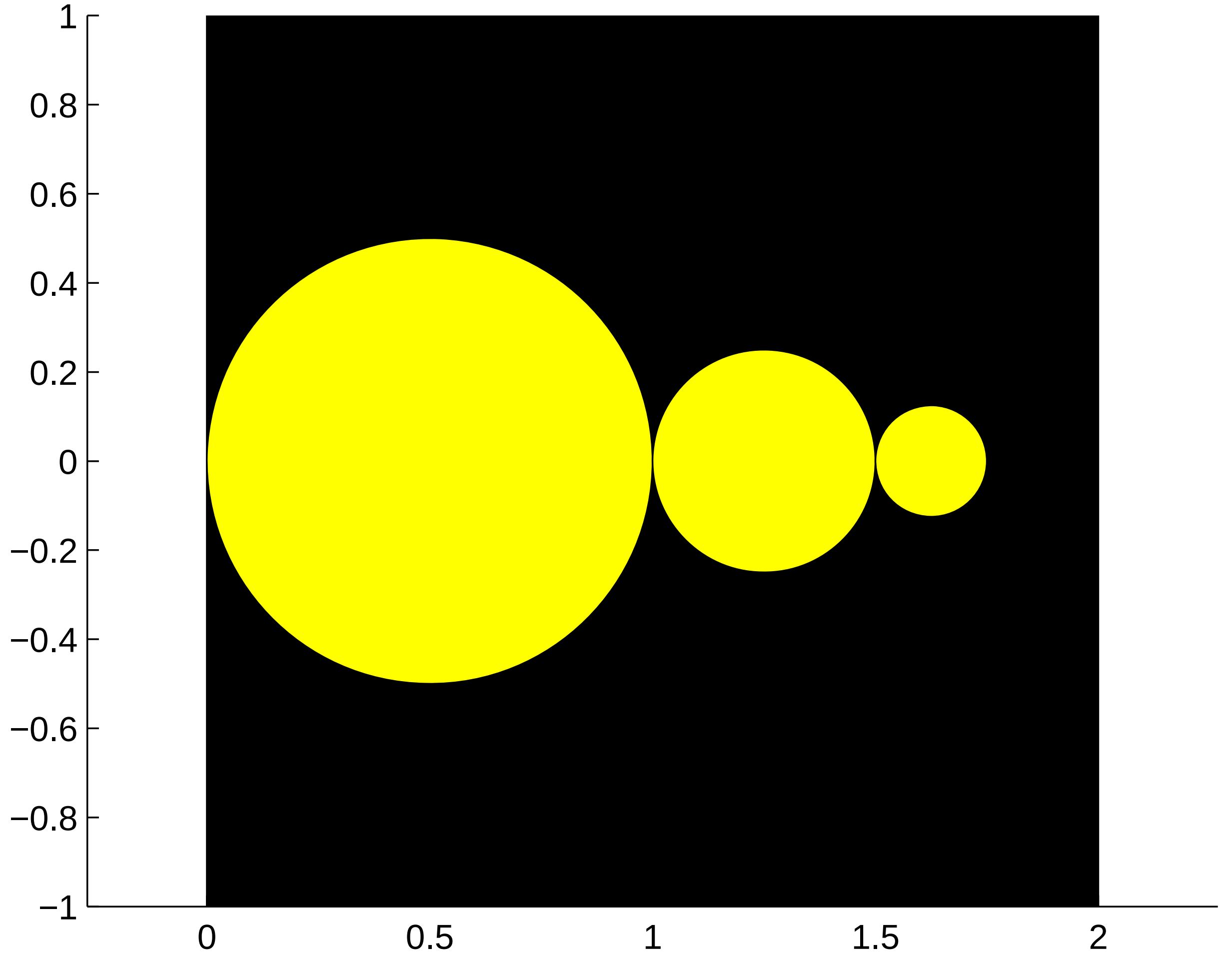
9. The Discrete
vs
The Continuous

Finite Arithmetic

More practice with iteration and conditionals.

Screen Granularity



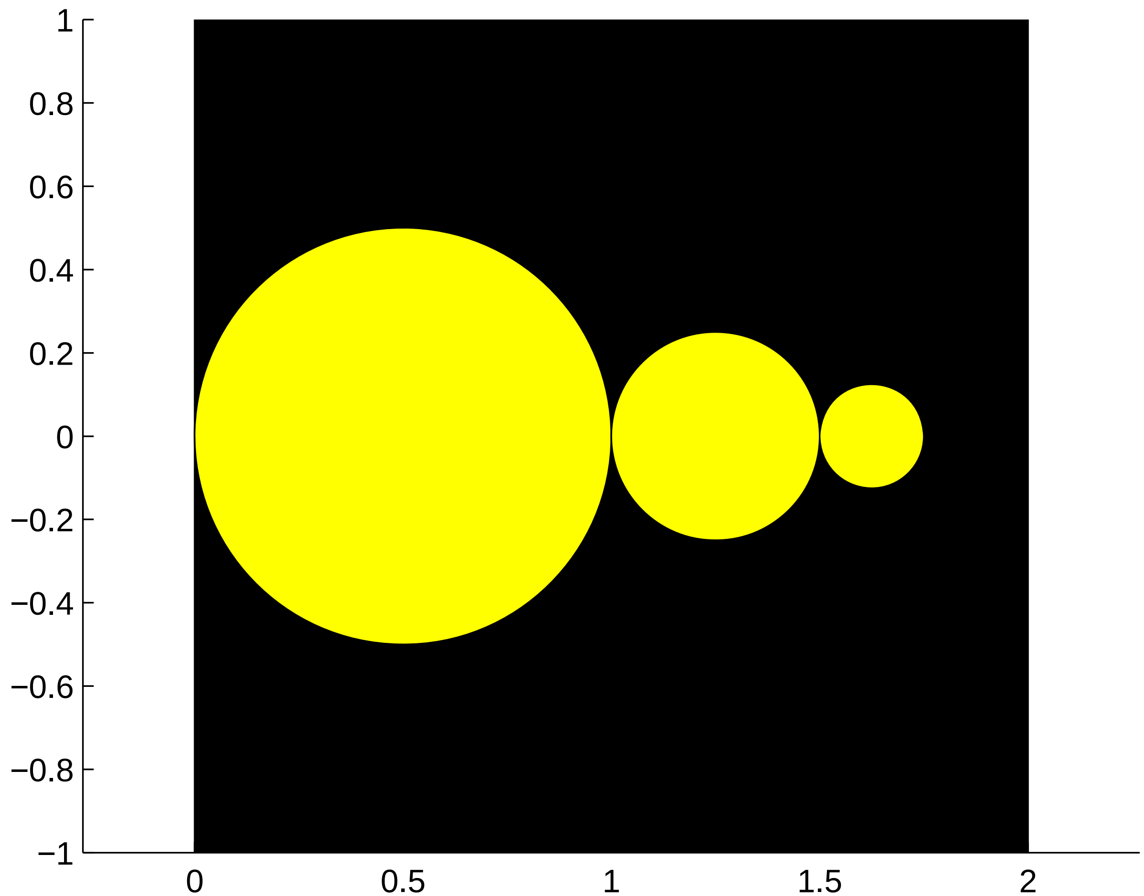
After how many halvings will the disks disappear?

Xeno's Paradox

- A wall is two feet away.
- Take steps that repeatedly halve the remaining distance.
- You never reach the wall because the distance traveled after n steps =

$$1 + \frac{1}{2} + \frac{1}{4} + \ldots + 1/2^n = 2 - 1/2^n$$

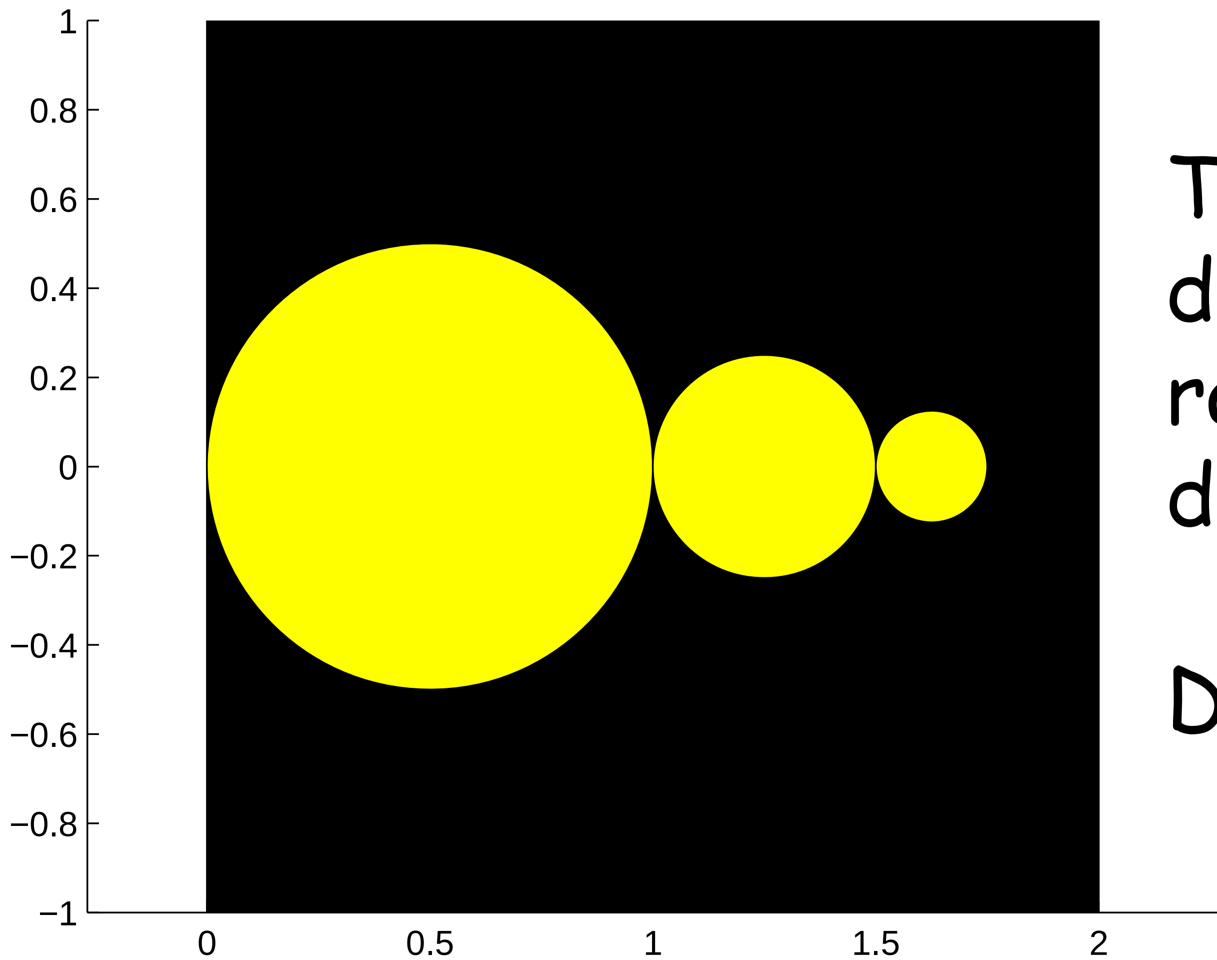
Problem: "Xeno" Disks



First disk has radius 1 and center (1/2 , 0).

The disks are tangent to each other and have centers on x-axis

Problem: Xeno Disks



The radius of a disk is half the radius of the disk to its left.

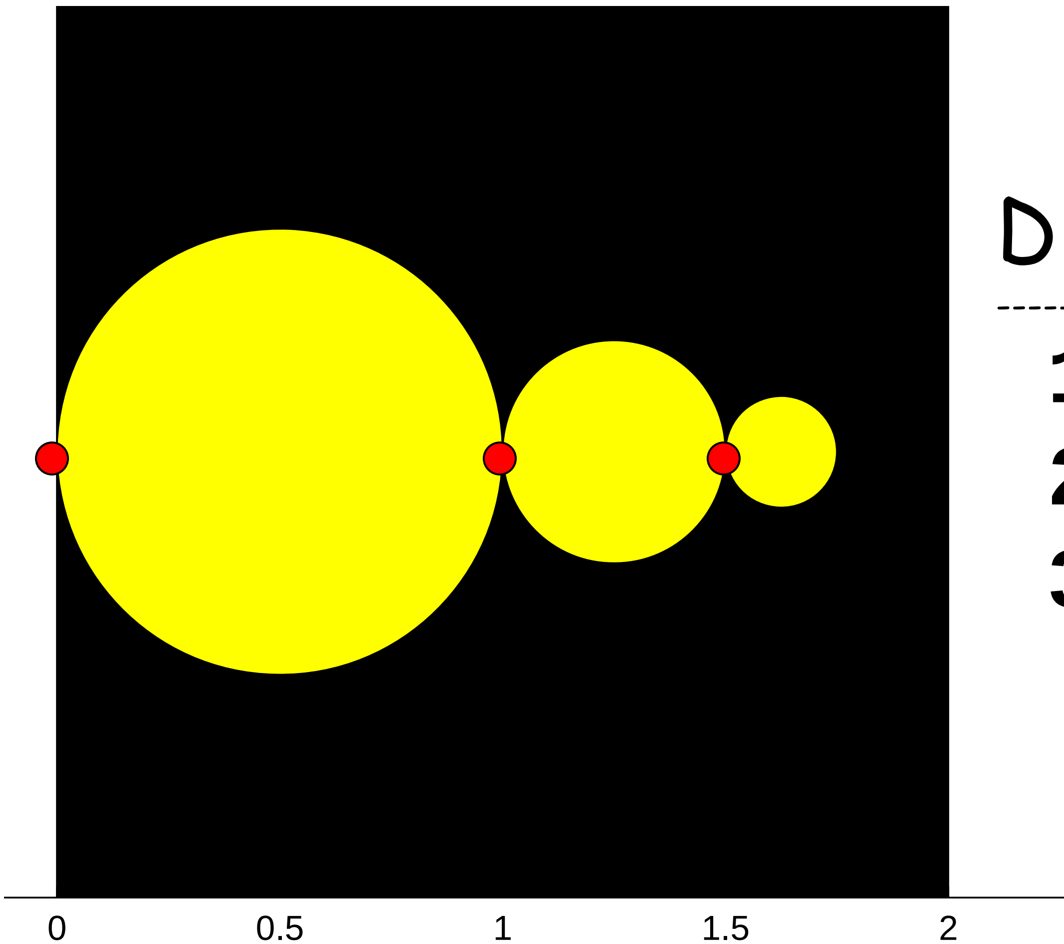
Draw 20 disks.

Variable Definitions

x: the x-value of the left tangent point for a given circle.

d : the diameter of a given circle

Preliminary Notes



Disk	x	d
1	0	1
2	0+1	1/2
3	0+1+1/2	1/4

Pseudocode

```
x = 0; d = 1
for k=1:20
```

Draw the next disk.
Update x and d.

```
end
```

Refinement

Draw the next disk

Draw disk with diameter d
and left tangent point (x,0)

```
DrawDisk(x+d/2,0,d/2,'y')
```

Refinement

Update x and d?

Disk	x	d
1	0	1
2	0+1	1/2
3	0+1+1/2	1/4

Next x is current x + current d.
Next d is one-half current d.

Refinement

Update x and d.

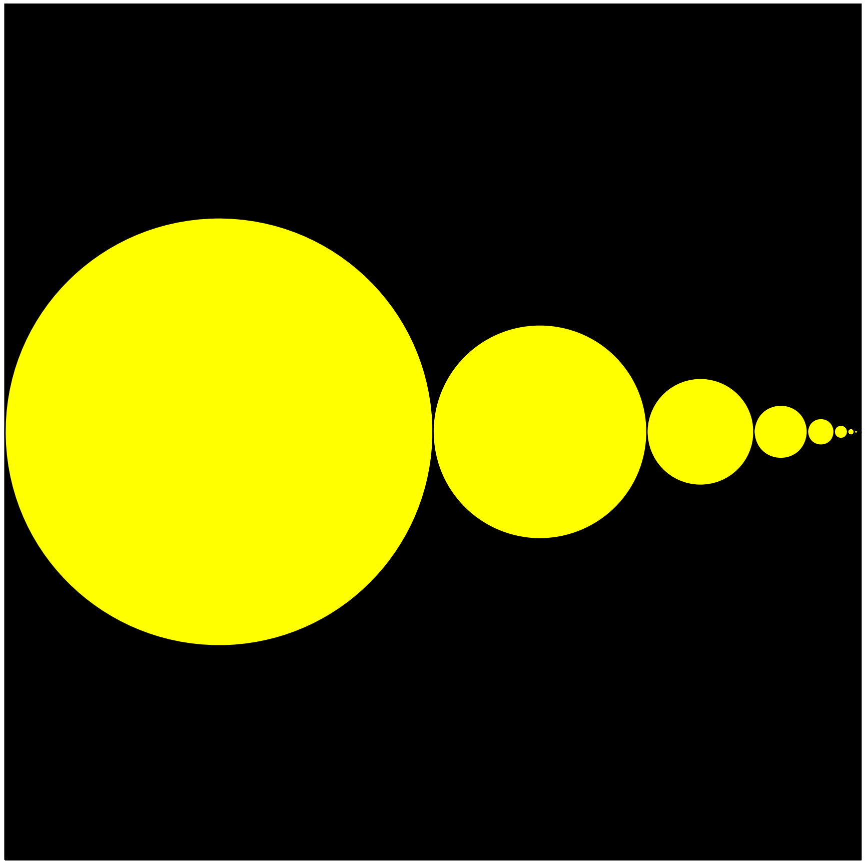
Next x is current x + current d.
Next d is one-half current d.

```
x = x + d;
d = d/2;
```

Solution

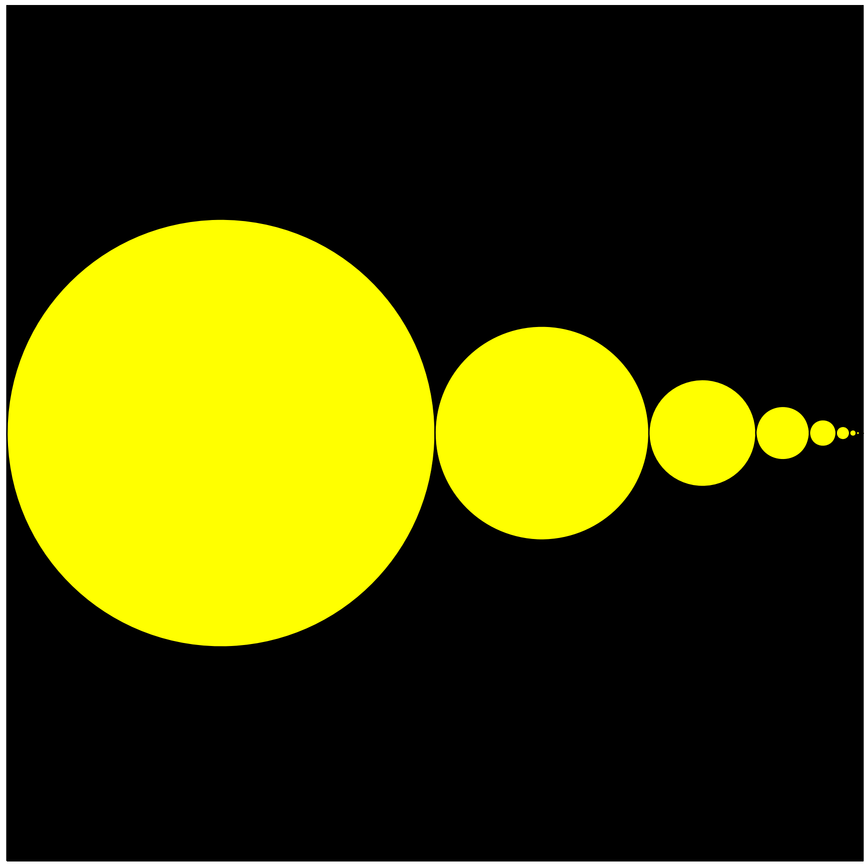
```
x = 0;
d = 1;
for k = 1:20
   DrawDisk(x+d/2,0,d/2,'y')
   x = x+d;
   d = d/2;
end
```

Output



Shouldn't there be 20 disks?

Screen is an Array of Dots*



*Called "Pixels"

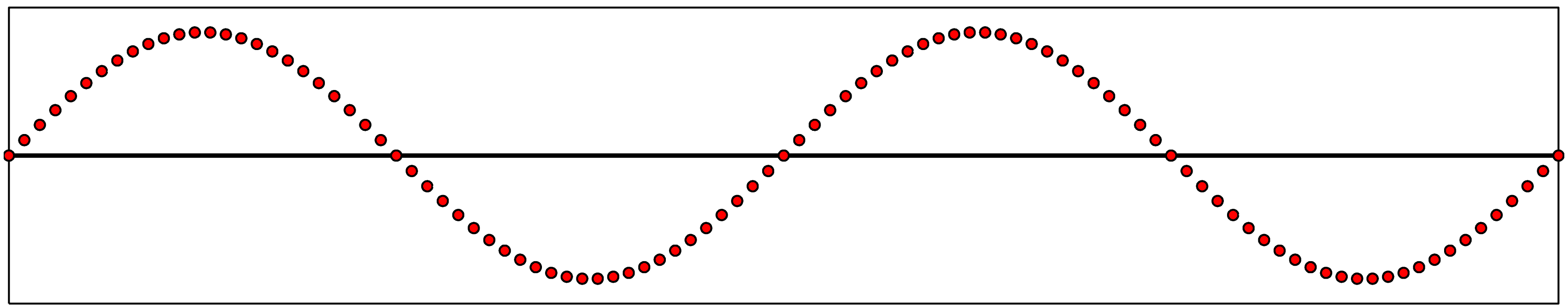
Disks smaller than the dots don't show up.

The 20th disk has radius < .000001

Finiteness

It shows up all over
the place in computing.

Plotting Continuous Functions



Can only display a bunch of dots

Another "collision" between the infinite and the finite. (More later.)

The Discrete Display of Sine

```
N = 100;
X_spacing = 4*pi/N;
Dot_radius = X_spacing/3;
for k=0:N
   x = k*X_spacing;
   y = sin(x);
   DrawDisk(x,y,Dot_Radius,'r')
end
```

The Moral

To produce realistic plots/renderings you must appreciate screen granularity.

Similar Finite "Behavior" with Computer Arithmetic

Memory Hardware is finite.

Computer cannot store never-ending decimals like pi, sqrt(2), 1/3.

Question Time

Does this script print anything?

```
k = 0;
while 1 + 1/2^k > 1
    k = k+1;
end
k = k
```

A. Yes   B. No   E. None of these

Similar "Behavior" for Computer Arithmetic

Suppose you have a calculator with a window like this:

+	2	4	1	-	3

Representing 2.41×10^{-3}

Add:

+	2	4	1	-	3

+	1	0	0	-	3

Result:

+	3	4	1	-	3

Add:

+	2	4	1	-	3

+	1	0	0	-	4

Result:

+	2	5	1	-	3

Add:

+	2	4	1	-	3

+	1	0	0	-	5

Result:

+	2	4	2	-	3

Add:

+	2	4	1	-	3

+	1	0	0	-	6

Result:

+	2	4	1	-	3

Add:

+	2	4	1	-	3

+	1	0	0	-	6

Result:

+	2	4	1	-	3

Not enough room to represent .002411

Regarding the Question…

The following loop does terminate and the concluding value of k that is displayed is 53.

```
k = 0;
while 1 + 1/2^k > 1
   k = k+1;
end
k = k
```

The Moral

To produce reliable numerical results you must appreciate floating point arithmetic.

The 1991 Patriot Missile Disaster

Elementary misperceptions about the finiteness of computer arithmetic. 30+ died.

The Setting

External clock counts time in tenths of seconds.

Targeting software needs time to compute trajectories. The method:

Time = (# external clock ticks) x (1/10)

The problem is here

One-Tenth in Binary

Exact:

.00011001100110011001100110011…

Patriot System used:

$\boxed{\mathbf{.00011001100110011001100}}$**110011…**

Error = .000000095sec every clock tick

Error

Time = (# external clock ticks) x (1/10)

Error = (# external clock ticks) x (.000000095)

After 100 hours...

Error = (100x60x60*10)*.000000095

= .34 secs

Missed target by 500 meters.