Previous Lecture:
- Examples on vectors and simulation

Today’s Lecture:
- Finite vs. Infinite; Discrete vs. Continuous
- Vectors and vectorized code
- Color computation with linear interpolation

Announcements:
- Project 3 due Monday at 11pm
- Prelim 1 on Thursday 10/13 at 7:30pm. You must notify us now if you have an exam conflict. Email Randy Hess (rbh27) with your conflict information (course number, instructor contact info).
Loop patterns for working with a vector

% Given a vector v
for k = 1:length(v)
    % Work with v(k)
    % E.g., disp(v(k))
end

% Given a vector v
k = 1;
while k<=length(v)
    % Work with v(k)
    % E.g., disp(v(k))
    k = k+1;
end
Discrete vs. continuous

A plot is made from discrete values, but it can look continuous if there're many points.
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 =

\[ 1 + \frac{1}{2} + \frac{1}{4} + \ldots + \frac{1}{2^n} = 2 - \frac{1}{2^n} \]
Example: “Xeno” disks

Draw a sequence of 20 disks where the \((k+1)\)th disk has a diameter that is half that of the \(k\)th disk.

The disks are tangent to each other and have centers on the \(x\)-axis.

First disk has diameter 1 and center \((1/2, 0)\).
Example: “Xeno” disks

What do you need to keep track of?

- Diameter \((d)\)
- Position
- Left tangent point \((x)\)

<table>
<thead>
<tr>
<th>Disk</th>
<th>(x)</th>
<th>(d)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0+1</td>
<td>1/2</td>
</tr>
<tr>
<td>3</td>
<td>0+1+1/2</td>
<td>1/4</td>
</tr>
</tbody>
</table>
% Xeno Disks

DrawRect(0,-1,2,2,'k')

% Draw 20 Xeno disks
% Xeno Disks

DrawRect(0,-1,2,2,'k')
% Draw 20 Xeno disks

for k= 1:20

end
% Xeno Disks

DrawRect(0,-1,2,2,'k')
% Draw 20 Xeno disks

d= 1;
x= 0;  % Left tangent point

for k= 1:20

end
% Xeno Disks

DrawRect(0,-1,2,2,'k')
% Draw 20 Xeno disks
d= 1;
x= 0;  % Left tangent point
for k= 1:20
    % Draw kth disk

    % Update x, d for next disk

end
% Xeno Disks

DrawRect(0,-1,2,2,'k')
% Draw 20 Xeno disks

d= 1;
x= 0;  % Left tangent point
for k= 1:20
    % Draw kth disk
    DrawDisk(x+d/2, 0, d/2, 'y')
    % Update x, d for next disk
    x= x+d;
d= d/2;
end
Here’s the output…  Shouldn’t there be 20 disks?

The “screen” is an array of dots called pixels.

Disks smaller than the dots don’t show up.

The 20th disk has radius < .000001
Fading Xeno disks

- First disk is yellow
- Last disk is black (invisible)
- Interpolate the color in between
Color is a 3-vector, sometimes called the RGB values

- Any color is a mix of red, green, and blue
- Example:
  \[
  \text{colr}= [0.4 \ 0.6 \ 0]
  \]
- Each component is a real value in \([0,1]\)
- \([0 \ 0 \ 0]\) is black
- \([1 \ 1 \ 1]\) is white
% Draw n Xeno disks

d = 1;
x = 0;  % Left tangent point

for k = 1:n

% Draw kth disk
DrawDisk(x+d/2, 0, d/2, 'y')
x = x+d;
d = d/2;
end
% Draw n Xeno disks

d = 1;
x = 0;  % Left tangent point

for k = 1:n
    % Draw kth disk
    DrawDisk(x+d/2, 0, d/2, [1 1 0])
    x = x+d;
d = d/2;
end

A vector of length 3
% Draw n fading Xeno disks

\[ d = 1; \]
\[ x = 0; \quad \% \text{Left tangent point} \]
\[ \text{yellow} = [1 \ 1 \ 0]; \]
\[ \text{black} = [0 \ 0 \ 0]; \]

\texttt{for k = 1:n}

\hspace{1em} % Compute color of kth disk

\hspace{1em} % Draw kth disk

\hspace{1em} \text{DrawDisk}(x+d/2, 0, d/2, \_\_\_\_\_\_\_\_\_\_\_\_\_) \]
\hspace{1em} x = x+d;
\hspace{1em} d = d/2;

\texttt{end}
Example: 3 disks fading from yellow to black

\[ r = 1; \quad % \text{radius of disk} \]
\[ \text{yellow} = [1 \ 1 \ 0]; \]
\[ \text{black} = [0 \ 0 \ 0]; \]

% Left disk yellow, at x=1
\text{DrawDisk}(1,0,r,\text{yellow})

% Right disk black, at x=5
\text{DrawDisk}(5,0,r,\text{black})

% Middle disk with average color, at x=3
\text{colr} = 0.5*\text{yellow} + 0.5*\text{black};
\text{DrawDisk}(3,0,r,\text{colr})
Example: 3 disks fading from yellow to black

\[ r = 1; \quad \% \text{radius of disk} \]
\[ \text{yellow} = [1 \ 1 \ 0]; \]
\[ \text{black} = [0 \ 0 \ 0]; \]

\% Left disk yellow, at x=1
\text{DrawDisk}(1,0,r,\text{yellow})

\% Right disk black, at x=5
\text{DrawDisk}(5,0,r,\text{black})

\% Middle disk with average color, at x=3
\text{colr} = 0.5*\text{yellow} + 0.5*\text{black};
\text{DrawDisk}(3,0,r,\text{colr})
Example: 3 disks fading from yellow to black

\[
\begin{align*}
\text{r} &= 1; \quad \% \text{ radius of disk} \\
\text{yellow} &= [1 \ 1 \ 0]; \\
\text{black} &= [0 \ 0 \ 0]; \\
\% \text{ Left disk yellow, at x=1} \\
\text{DrawDisk}(1, 0, r, \text{yellow}) \\
\% \text{ Right disk black, at x=5} \\
\text{DrawDisk}(5, 0, r, \text{black}) \\
\% \text{ Middle disk with average color, at x=3} \\
\text{colr} &= 0.5*\text{yellow} + 0.5*\text{black}; \\
\text{DrawDisk}(3, 0, r, \text{colr}) \quad \text{Vectorized addition} \\
\end{align*}
\]
Vectorized code allows an operation on multiple values at the same time

\[ \text{yellow} = [1 \ 1 \ 0]; \]
\[ \text{black} = [0 \ 0 \ 0]; \]

\% Average color via \textit{vectorized} op
\[ \text{colr} = 0.5^{*}\text{yellow} + 0.5^{*}\text{black}; \]

\% Average color via \textit{scalar} op
\[
\text{for } k = 1: \text{length}(\text{black}) \\
\quad \text{colr}(k) = 0.5^{*}\text{yellow}(k) + 0.5^{*}\text{black}(k); \\
\text{end}
\]

\[
\begin{align*}
\begin{array}{cccc}
0.5 & 0.5 & 0 \\
0 & 0 & 0
\end{array}
\end{align*}
\]

\[
\begin{align*}
\begin{array}{cccc}
0.5 & 0.5 & 0 \\
\end{array}
\end{align*}
\]
% Draw n fading Xeno disks

d = 1;
x = 0;  % Left tangent point
yellow = [1 1 0];
black = [0 0 0];

for k = 1:n
    % Compute color of kth disk

    % Draw kth disk
    DrawDisk(x+d/2, 0, d/2, ______)
x = x+d;
d = d/2;
end
% Draw n fading Xeno disks

d= 1;
x= 0;  % Left tangent point
yellow= [1 1 0];
black= [0 0 0];

for k= 1:n

   % Compute color of kth disk
   colr= ___*black + ___*yellow;

   % Draw kth disk
   DrawDisk(x+d/2, 0, d/2, colr)
   x= x+d;
   d= d/2;

end
Use *linear interpolation* to obtain the colors. Each disk has a color that is a linear combination of yellow and black. Let $f$ be a fraction in $(0,1)$ ...

\[
f = \ ??? \]
\[
colr = f \cdot \text{black} + (1-f) \cdot \text{yellow};
\]
Linear interpolation

<table>
<thead>
<tr>
<th>x</th>
<th>g(x)</th>
</tr>
</thead>
<tbody>
<tr>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>9</td>
<td>110</td>
</tr>
<tr>
<td>10</td>
<td>118</td>
</tr>
<tr>
<td>11</td>
<td>126</td>
</tr>
<tr>
<td>12</td>
<td>134</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>
### Linear interpolation

<table>
<thead>
<tr>
<th>x</th>
<th>g(x)</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>110</td>
</tr>
<tr>
<td>10</td>
<td>118</td>
</tr>
<tr>
<td>10.25</td>
<td>?</td>
</tr>
<tr>
<td>10.50</td>
<td>?</td>
</tr>
<tr>
<td>10.75</td>
<td>?</td>
</tr>
<tr>
<td>11</td>
<td>126</td>
</tr>
<tr>
<td>12</td>
<td>134</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ g(10.5) = \left[ \frac{g(11) + g(10)}{2} \right] / 2 \]
Linear interpolation

<table>
<thead>
<tr>
<th>x</th>
<th>g(x)</th>
</tr>
</thead>
<tbody>
<tr>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>9</td>
<td>110</td>
</tr>
<tr>
<td>10</td>
<td>118</td>
</tr>
<tr>
<td>10.25</td>
<td>?</td>
</tr>
<tr>
<td>10.50</td>
<td>?</td>
</tr>
<tr>
<td>10.75</td>
<td>?</td>
</tr>
<tr>
<td>11</td>
<td>126</td>
</tr>
<tr>
<td>12</td>
<td>134</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

\[
g(10.5) = \frac{1}{2} g(11) + \frac{1}{2} g(10)
\]

\[
g(10.25) = \frac{1}{4} g(11) + \frac{3}{4} g(10)
\]

\[
g(10.50) = \frac{2}{4} g(11) + \frac{2}{4} g(10)
\]

\[
g(10.75) = \frac{3}{4} g(11) + \frac{1}{4} g(10)
\]
Linear interpolation

<table>
<thead>
<tr>
<th>x</th>
<th>g(x)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>110</td>
</tr>
<tr>
<td>10</td>
<td>118</td>
</tr>
<tr>
<td>10.25</td>
<td>?</td>
</tr>
<tr>
<td>10.50</td>
<td>?</td>
</tr>
<tr>
<td>10.75</td>
<td>?</td>
</tr>
<tr>
<td>11</td>
<td>126</td>
</tr>
<tr>
<td>12</td>
<td>134</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[
g(10.5) = \frac{1}{2} g(11) + \frac{1}{2} g(10)
\]

\[
g(10) = \frac{0}{4} \cdot g(11) + \frac{4}{4} \cdot g(10)
\]

\[
g(10.25) = \frac{1}{4} \cdot g(11) + \frac{3}{4} \cdot g(10)
\]

\[
g(10.50) = \frac{2}{4} \cdot g(11) + \frac{2}{4} \cdot g(10)
\]

\[
g(10.75) = \frac{3}{4} \cdot g(11) + \frac{1}{4} \cdot g(10)
\]

\[
g(11) = \frac{4}{4} \cdot g(11) + \frac{0}{4} \cdot g(10)
\]

\[
f \cdot g(11) + (1-f) \cdot g(10)
\]
% Draw n fading Xeno disks

d= 1;
x= 0;  % Left tangent point
yellow= [1 1 0];
black= [0 0 0];

for k= 1:n

  % Compute color of kth disk
  f= ???
  colr= f*black + (1-f)*yellow;

  % Draw kth disk
  DrawDisk(x+d/2, 0, d/2, colr)

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

end
Rows of Xeno disks

for y = __ : __ : __

Code to draw one row of Xeno disks at some y-coordinate

end

Be careful with initializations
yellow=[1 1 0];  black=[0 0 0];

d= 1;

x= 0;

for k= 1:n
    \% Compute color of kth disk
    f= (k-1)/(n-1);
    colr= f*black + (1-f)*yellow;
    \% Draw kth disk
    DrawDisk(x+d/2, 0, d/2, colr)
    x=x+d;  d=d/2;
end
yellow=[1 1 0];  black=[0 0 0];

d= 1;

x= 0;

for k= 1:n
    % Compute color of kth disk
    f= (k-1)/(n-1);
    colr= f*black + (1-f)*yellow;
    % Draw kth disk
    DrawDisk(x+d/2, 0, d/2, colr)
    x=x+d;  d=d/2;

end
yellow=[1 1 0]; black=[0 0 0];
for y= ___:___:___
d= 1;  

% Computations necessary for each row
x= 0;

for k= 1:n
% Compute color of kth disk
f= (k-1)/(n-1);
colr= f*black + (1-f)*yellow;
% Draw kth disk
DrawDisk(x+d/2, 0, d/2, colr)
x=x+d;  d=d/2;
end
end
Does this script print anything?

```plaintext
k = 0;
while 1 + 1/2^k > 1
    k = k+1;
end
disp(k)
```
Computer Arithmetic—floating point arithmetic

Suppose you have a calculator with a window like this:

\[
\begin{array}{cccccc}
+ & 2 & 4 & 1 & - & 3 \\
\end{array}
\]

representing \( 2.41 \times 10^{-3} \)
Floating point addition

```
+ 2 4 1 - 3
```

```
+ 1 0 0 - 3
```

Result:

```
+ 3 4 1 - 3
```
Floating point addition

\[ +241 -3 \]

\[ +100 -4 \]

Result: \[ +251 -3 \]
Floating point addition

\[
\begin{align*}
+ & \quad 2 & 4 & 1 & \quad - & \quad 3 \\
+ & \quad 1 & 0 & 0 & \quad - & \quad 5 \\
\end{align*}
\]

Result: \[
\begin{align*}
+ & \quad 2 & 4 & 2 & \quad - & \quad 3 \\
\end{align*}
\]
Floating point addition

\[ +\quad 2\quad 4\quad 1 \quad - \quad 3 \]
\[ +\quad 1\quad 0\quad 0 \quad - \quad 6 \]
\[ \text{Result:} \quad +\quad 2\quad 4\quad 1 \quad - \quad 3 \]

Not enough room to represent \(.002411\)
The loop DOES terminate given the limitations of floating point arithmetic!

```matlab
k = 0;
while 1 + 1/2^k > 1
    k = k+1;
end
disp(k)
```

1 + 1/2^53 is calculated to be just 1, so “53” is printed.