CSE 8A Lecture 11

• In-term exam 3: Thursday. YOU WILL WRITE CODE to manipulate an image, from scratch. Study all of the image manipulation code you’ve seen so far, and practice writing these methods from scratch.

• PSA6 due tonight, 11:59pm
  – Sorry about submission issues… Hopefully won’t happen again. But if it does, you are allowed to do the interview before submitting (you’ll anyway need to show the code to tutor)

• Today:
  – Binary numbers
  – Steganography!
  – Some exam practice-problems at the end of the slides!
How can we hide the bunny in the hat?
Cathy saw eight (8) airplanes penetrate skies aloft. 7 really only could keep straight.
Steganography: Hiding information in plain sight

Cathy saw eight (8) airplanes penetrate skies aloft. 7 really only could keep straight.
A German press cable from WW I

President's embargo ruling should have immediate notice. Grave situation affecting international law. Statement foreshadows ruin of many neutrals. Yellow journals unifying national excitement immensely.
President's embargo ruling should have immediate notice. Grave situation affecting international law. Statement foreshadows ruin of many neutrals. Yellow journals unifying national excitement immensely. 

_Pershing sails from N.Y. June 1._
And now for an equally bad approach…

Are these two colors the same, or different?

A. The same
B. Different
### Hiding information in images

<table>
<thead>
<tr>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>39</td>
<td>56</td>
<td>101</td>
</tr>
</tbody>
</table>

Write each of these color values as an 8-bit binary number:

```
( 00100101, 01110010, 11110001 )
```
Hiding information in images

<table>
<thead>
<tr>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>39</td>
<td>56</td>
<td>101</td>
</tr>
</tbody>
</table>

Write each of these color values as an 8-bit binary number

( 00100111, 00111000, 01100101 )
Hiding information in images

<table>
<thead>
<tr>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>(39,)</td>
<td>56,</td>
<td>101)</td>
</tr>
<tr>
<td>(37,)</td>
<td>59,</td>
<td>100)</td>
</tr>
</tbody>
</table>

Will the second color above look the same, or different?
A. Same  B. Different
(00100111, 00111000, 01100101)
Hiding information in images

<table>
<thead>
<tr>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>(39, 56, 101)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(37, 59, 100)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What is the *maximum* amount we can change a value by changing its two least significant digits?

A. 1    B. 2    C. 3    D. 4

(00100111, 00111000, 01100101)

(00100101, 00111011, 01100100)
### Hiding information in images

<table>
<thead>
<tr>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>(39,</td>
<td>56,</td>
<td>101)</td>
</tr>
<tr>
<td>(37,</td>
<td>59,</td>
<td>100)</td>
</tr>
</tbody>
</table>

What is the *maximum* amount we can change a value by changing its two least significant digits?

A. 1  
B. 2  
C. 3  
D. 4

(00100111, 00111000, 01100101)

(00100101, 00111011, 01100100)

No matter how we change the last two bits, the color will still look the same. (This is probably true for the last 3 bits too…)
Hiding information in images

<table>
<thead>
<tr>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>(39,</td>
<td>56,</td>
<td>101)</td>
</tr>
<tr>
<td>(37,</td>
<td>59,</td>
<td>100)</td>
</tr>
</tbody>
</table>

What is the maximum amount we can change a value (in decimal) by changing its three least significant digits?

A. 1       B. 3       C. 4       D. 7       E. 8

(00100\textcolor{red}{111}, 00111\textcolor{red}{000}, \textcolor{red}{01100101})

(00100\textcolor{red}{101}, 00111\textcolor{red}{011}, \textcolor{red}{01100100})
Hiding information in images

<table>
<thead>
<tr>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>(39, 56, 101)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(37, 59, 100)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

What is the maximum amount we can change a value (in decimal) by changing its three least significant digits?

A. 1  
B. 3  
C. 4  
D. 7  
E. 8

\( \begin{align*}
(00100111, & \ 00111000 \ , \ 01100101) \\
(00100101, & \ 00111011 \ , \ 01100100) \\
\end{align*}\)

Conclusion: We can do whatever we want with the two (and probably three, maybe four?) least significant bits in each color channel without changing the visual appearance of the image. But how does this help us…?
Degraded color information

Using 8 bits per color gives us how many possible values for each color channel?
A. 8  B. 255  C. 256  D. $2^8 - 1$

(00100111, 00111000, 01100101)
Degraded color information

Using 8 bits per color gives us how many possible values for each color channel?
C. 256 (2^8)

( 00100111, 00111000, 01100101 )

How many different colors does this allow us to represent?
A. 256       B. 256 * 3       C. 256^3
Using 2 bits per color gives us how many possible values for each color channel?

A. 2  B. 3  C. 4  D. 8

(11, 00, 01)
Degraded color information

Using 2 bits per color gives us how many possible values for each color channel?
C. 4 \( (2^2) \)

\[ (11, 00, 01) \]

How many different colors does this allow us to represent?
A. 4  B. 4*3  C. 4^3
Degraded color information

Using 2 bits per color gives us how many possible values for each color channel?
C. 4 \( (2^2) \)

\( \begin{align*}
( & 11, & 00, & 01 )
\end{align*} \)

How many different colors does this allow us to represent?
A. 4  B. 4*3  C. 4^3
2-bit color representation

(11, 00, 01)

What color should this be?
A. Pink(ish)
B. Blue
C. Green
D. White
E. Black
2-bit color representation

(11, 00, 01)

A lot of red

No green

A little blue
2-bit color representation

(11, 00, 01)

What color will this be (in Java)?
A. Pink(ish)
B. Blue
C. Green
D. White
E. Black
Bit shifting

Most significant digit ($2^7$ place)  Least significant digit ($2^0$ place)
Bit shifting

What happens to the value of this number (3) if I shift it left one position?

0 0 0 0 0 0 1 1

Most significant digit (2^7 place)  Least significant digit (2^0 place)
What is the new value of the number (formerly 3)?
A. 6   B. 8   C. 11   D. 12   E. 42

Most significant digit (2^7 place)  Least significant digit (2^0 place)
Bit shifting

In general, what happens to a number each time you shift it left? Why?

Most significant digit (2^7 place)  Least significant digit (2^0 place)
Bit shifting

In general, what happens to a number each time you shift it left? Why?

Most significant digit (2^7 place)  Least significant digit (2^0 place)
Bit shifting

In general, what happens to a number each time you shift it left? Why?

Most significant digit (2^7 place)     Least significant digit (2^0 place)
In general, what happens to a number each time you shift it left? Why?

Most significant digit (2^7 place)  Least significant digit (2^0 place)
In general, what happens to a number each time you shift it left? Why?

Most significant digit ($2^7$ place)  Least significant digit ($2^0$ place)
Bit shifting

In general, what happens to a number each time you shift it left? Why?

| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |

Most significant digit (2^7 place)  Least significant digit (2^0 place)

A. Every time you shift left you multiply the number by 2  
B. Every time you shift left you square the number  
C. Every time you shift left you add 2 to the number  
D. Every time you shift left, the number doesn’t change
Bit shifting in Java

Most significant digit ($2^7$ place)  Least significant digit ($2^0$ place)

$>>> 6 \lll 2$

Left shift operator
“Shift the value 6 to the left twice”
Will yield $6 \times 2 \times 2 = 24$

DEMO in Dr. Java
Bit shifting in Java

You could write a loop to do the same thing, but why?

```
>>> 6 << 2
```

Left shift operator

“Shift the value 6 to the left 2 times”
Will yield $6 \times 2 \times 2 = 24$
Bit shifting in Java

Most significant digit (2^7 place)

Least significant digit (2^0 place)

>>> 6 >> 2

Right shift operator
“Shift the value 6 to the right 2 times”
Will yield (6 / 2) / 2 = 1
Hiding information in images

Pixel in context image
Red: 00100111, Green: 00111000, Blue: 01100101

Pixel in secret message (at same x, y position)
Red: 01100100, Green: 11111001, Blue: 00001111

SLIGHTLY DIFFERENT ALGORITHM FROM THE VIDEO
Hiding information in images

Pixel in context image: \((00100111, 00111000, 01100101)\)

Pixel in secret message (at same x, y position): \((01100100, 11111001, 00001111)\)

Turn the 8-bit secret message color into a 2-bit color by preserving the two most significant bits. (How? Why?)

Context image

Secret message
Hiding information in images

Pixel in context image  (00100111, 00111000, 01100101)

Pixel in secret message (at same x, y position)  (01, 11, 00)

Overwrite the two least significant digits in the context image with the 2-bit color of the secret message. (How?)
Hiding information in images

Pixel in context image
( 00100101, 00111011, 01100100 )

Pixel in secret message
(at same x, y position)
( 01, 11, 00 )

The secret message is now hidden in the context image! (Make a copy first!)
Recovering information from images

Pixel in context with secret message
(00100101, 00111011, 01100100)

Pixel in recovered message
(at same x, y position)
(01, 11, 00)

Find the two least significant digits from the image with the secret message (How?)
Recovering information from images

Pixel in context with secret message

Red: ( 00100101, 00111011, 01100100 )
Green: ( 01000000, 11000000, 00000000 )
Blue: ( 01000000, 11000000, 00000000 )

Pixel in recovered message (at same x, y position)

Shift the bits in the recovered message to the left. You’ve recovered the message!
Exam 3 practice problem

- Write a method in the Picture class that draws a 10x10 red square in the center of the calling object.

- Do this both by looping over the whole image and using an if statement, and by looping over ONLY the specified region.
Exam 3 practice problem

• Write a method in the Picture class that takes a threshold value for green (an int). It then creates a new Picture and copies into this new Picture only those pixels whose green value is above the threshold. It leaves the rest of the Pixels in the new Pictures blank. It returns the new Picture.
Exam 3 practice problem

• Write a method in the Picture class that takes a target Picture object and copies the calling object’s picture _upside down_ onto the target picture. Can you handle the case where the target is smaller than then calling object? How about where the target is larger?
• Reading for next class: none. Study for the exam.
• Finish on PSA6
• Feedback on the Piazza form 😊