Practice Exam 5 minutes

No cellphones allowed
No talking to your group

Today

- Ask me questions about PA3
- Work with Laptops
- Chapter 4
  - Repetitions

CS 101 Repetition

Chapter 4

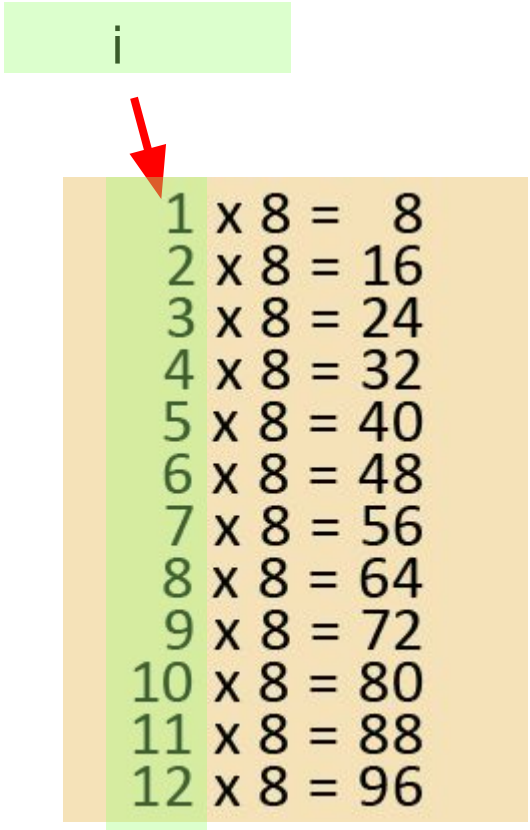
Last ICA - Question 5.

Write a program that prints the multiplication table of 8

1 x 8 = 8
2 x 8 = 16
3 x 8 = 24
4 x 8 = 32
5 x 8 = 40
6 x 8 = 48
7 x 8 = 56
8 x 8 = 64
9 x 8 = 72
10 x 8 = 80
11 x 8 = 88
12 x 8 = 96

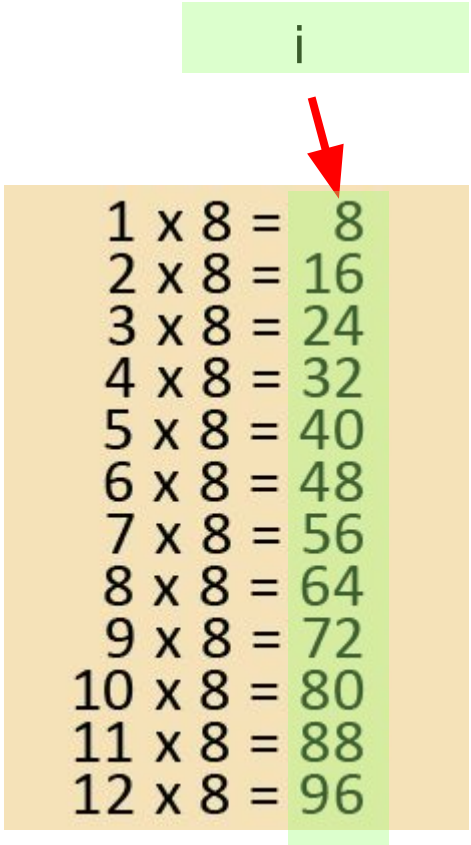
Solution 1. **i is the row**

```
int result = 0;
for (int i = 1; i<= 12; i +=1){
  result = i*8;
  println ( i , " x 8 = ", result);
}
```



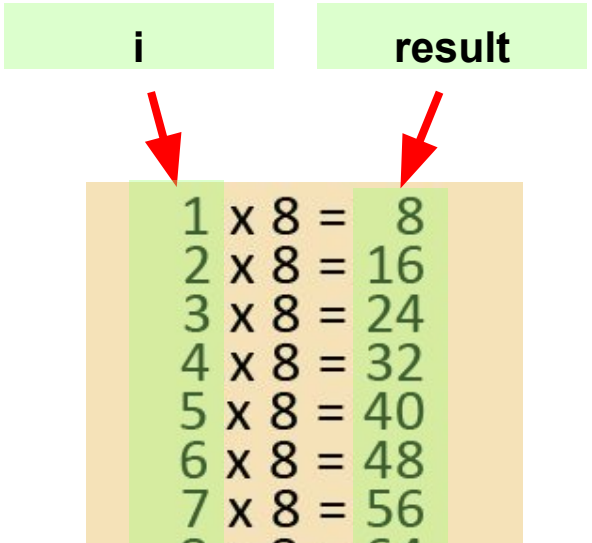
Solution 2. **i is the result**

```
int row = 0;
for (int i = 8; i<= 96; i += 8){
  row = i/8;
  println(row , " x 8 = ", i);
}
```



Solution 3. **i and result**

```
for(int i = 1, result = 8; i<=12; i+=1, result +=8) {
  println(i , " x 8 =", result) ;
}
```



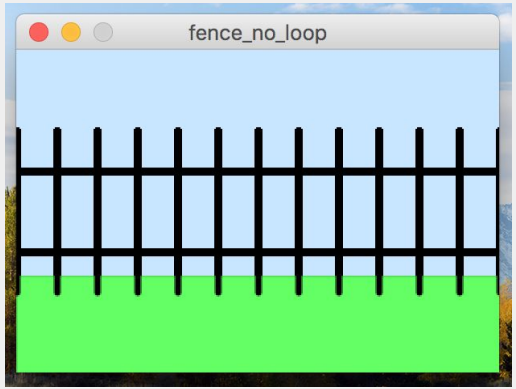
The loop test

- There is actually a lot of flexibility in the condition (or ***test***) in a for-loop
- It can compare any two values, not just the loop variable
- You can also use many types of comparisons
  - `>` greater-than
  - `<` less-than
  - `>=` greater-than or equal-to
  - `<=` less-than or equal-to
  - `==` exactly equal-to
  - `!=` not equal-to

Repeating Drawings

Back to Fence-Post

- Now that we have a little practice with loops, let's simplify the code




```
void setup() {
  size(300, 200);
  background(200, 230, 255);
}

void draw() {
  strokeWeight(0);
  fill(100, 255, 100);
  rect(0, 140, 300, 100);
  strokeWeight(4);
  line(  0,  50,  0, 150);
  line( 25,  50, 25, 150);
  line( 50,  50, 50, 150);
  line( 75,  50, 75, 150);
  line(100, 50, 100, 150);
  line(125, 50, 125, 150);
  line(150, 50, 150, 150);
  line(175, 50, 175, 150);
  line(200, 50, 200, 150);
  line(225, 50, 225, 150);
  line(250, 50, 250, 150);
  line(275, 50, 275, 150);
  line(300, 50, 300, 150);
  line(0, 75, 300, 75);
  line(0, 125, 300, 125);
}
```

Simplify the fence-post code

- How can we use one (or more) for-loops to simplify this code?
- Discuss in your groups

```
void setup() {
  size(300, 200);
  background(200, 230, 255);
}

void draw() {
  strokeWeight(0);
  fill(100, 255, 100);
  rect(0, 140, 300, 100);
  strokeWeight(4);
  line(  0,  50,  0, 150);
  line( 25,  50, 25, 150);
  line( 50,  50, 50, 150);
  line( 75,  50, 75, 150);
  line(100, 50, 100, 150);
  line(125, 50, 125, 150);
  line(150, 50, 150, 150);
  line(175, 50, 175, 150);
  line(200, 50, 200, 150);
  line(225, 50, 225, 150);
  line(250, 50, 250, 150);
  line(275, 50, 275, 150);
  line(300, 50, 300, 150);
  line(0, 75, 300, 75);
  line(0, 125, 300, 125);
}
```

Identifying Repetition

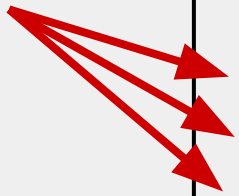
Need to identify:

- What shape(s) are repeatedly drawn?
- What changes from shape to shape? (x, y, width, height, color, etc)
- By how much does the change occur?

```
void setup() {
  size(300, 200);
  background(200, 230, 255);
}

void draw() {
  strokeWeight(0);
  fill(100, 255, 100);
  rect(0, 140, 300, 100);
  strokeWeight(4);
  line(  0,  50,  0, 150);
  line( 25,  50, 25, 150);
  line( 50,  50, 50, 150);
  line( 75,  50, 75, 150);
  line(100, 50, 100, 150);
  line(125, 50, 125, 150);
  line(150, 50, 150, 150);
  line(175, 50, 175, 150);
  line(200, 50, 200, 150);
  line(225, 50, 225, 150);
  line(250, 50, 250, 150);
  line(275, 50, 275, 150);
  line(300, 50, 300, 150);
  line(0, 75, 300, 75);
  line(0, 125, 300, 125);
}
```

Identifying Repetition

The vertical lines of the fence are the repeated shape

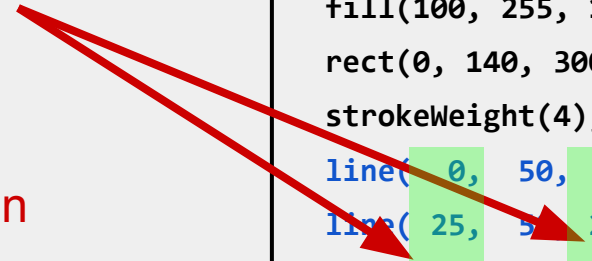


```
void setup() {
  size(300, 200);
  background(200, 230, 255);
}

void draw() {
  strokeWeight(0);
  fill(100, 255, 100);
  rect(0, 140, 300, 100);
  strokeWeight(4);
  line(  0,  50,  0, 150);
  line( 25,  50, 25, 150);
  line( 50,  50, 50, 150);
  line( 75,  50, 75, 150);
  line(100, 50, 100, 150);
  line(125, 50, 125, 150);
  line(150, 50, 150, 150);
  line(175, 50, 175, 150);
  line(200, 50, 200, 150);
  line(225, 50, 225, 150);
  line(250, 50, 250, 150);
  line(275, 50, 275, 150);
  line(300, 50, 300, 150);
  line(0, 75, 300, 75);
  line(0, 125, 300, 125);
}
```

Identifying Repetition

The x values are changing, the y values are not

Change happens in increments of 25



```
void setup() {
  size(300, 200);
  background(200, 230, 255);
}

void draw() {
  strokeWeight(0);
  fill(100, 255, 100);
  rect(0, 140, 300, 100);
  strokeWeight(4);
  line(  0,  50,  0, 150);
  line( 25,  50, 25, 150);
  line( 50,  50, 50, 150);
  line( 75,  50, 75, 150);
  line(100, 50, 100, 150);
  line(125, 50, 125, 150);
  line(150, 50, 150, 150);
  line(175, 50, 175, 150);
  line(200, 50, 200, 150);
  line(225, 50, 225, 150);
  line(250, 50, 250, 150);
  line(275, 50, 275, 150);
  line(300, 50, 300, 150);
  line(0, 75, 300, 75);
  line(0, 125, 300, 125);
}
```

Identifying Repetition

How can the same exact fence be drawn, using a for-loop to reduce the repetition?

```
void setup() {
  size(300, 200);
  background(200, 230, 255);
}

void draw() {
  strokeWeight(0);
  fill(100, 255, 100);
  rect(0, 140, 300, 100);
  strokeWeight(4);
  line(  0,  50,  0, 150);
  line( 25,  50, 25, 150);
  line( 50,  50, 50, 150);
  line( 75,  50, 75, 150);
  line(100, 50, 100, 150);
  line(125, 50, 125, 150);
  line(150, 50, 150, 150);
  line(175, 50, 175, 150);
  line(200, 50, 200, 150);
  line(225, 50, 225, 150);
  line(250, 50, 250, 150);
  line(275, 50, 275, 150);
  line(300, 50, 300, 150);
  line(0, 75, 300, 75);
  line(0, 125, 300, 125);
}
```

Identifying Repetition

```
void setup() {
  size(300, 200);
  background(200, 230, 255);
}

void draw() {
  strokeWeight(0);
  fill(100, 255, 100);
  rect(0, 140, 300, 100);
  strokeWeight(4);
  for(int i = 0; i < 301; i += 25) {
    line(i, 50, i, 150);
  }
  line(0, 75, 300, 75);
  line(0, 125, 300, 125);
}
```

Identifying Repetition

Modify the fence code to use a for loop to reduce the repetition!

Download the code (see the Lectures tab on the class website). Try:

- decreasing the spacing between the fenceposts
- increasing the spacing between the fenceposts

```
void setup() {
  size(300, 200);
  background(200, 230, 255);
}

void draw() {
  strokeWeight(0);
  fill(100, 255, 100);
  rect(0, 140, 300, 100);
  strokeWeight(4);
  for(int i = 0; i < 301; i += 25) {
    line(i, 50, i, 150);
  }
  line(0, 75, 300, 75);
  line(0, 125, 300, 125);
}
```

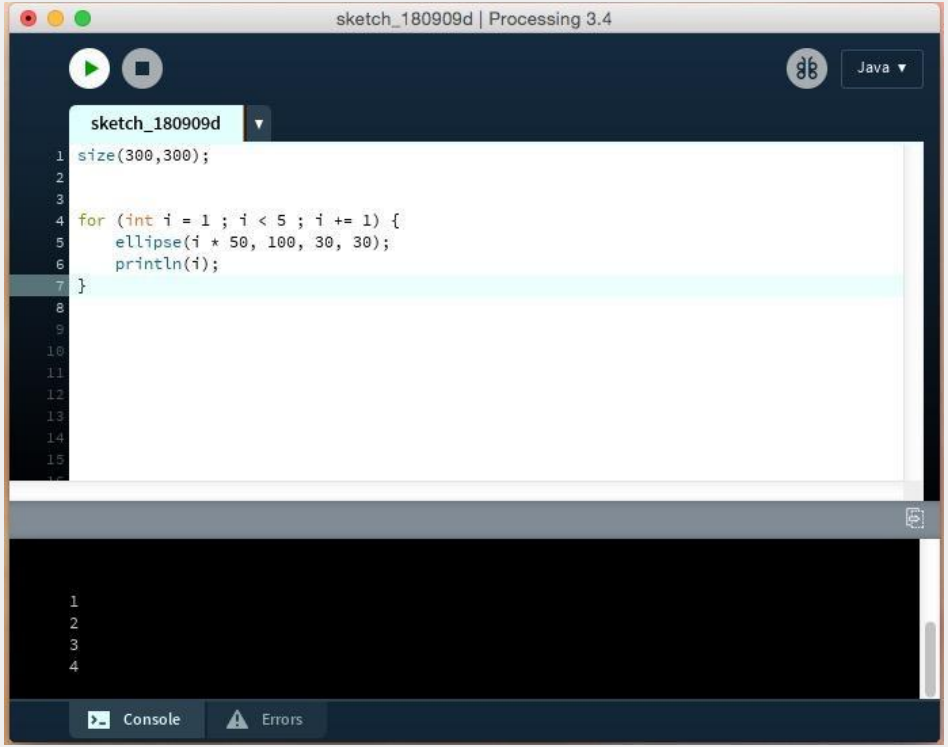
Explore the book examples (Chapter 4) and the following examples to solve the self-paced ICAs

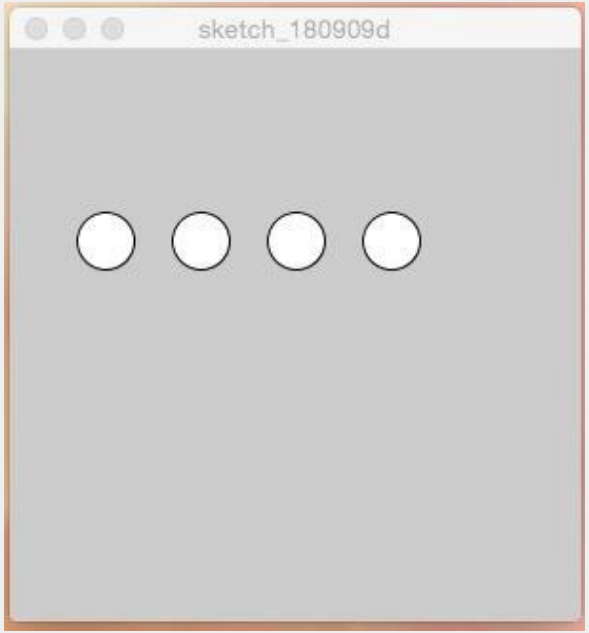
Example 1. for loop drawing

- Now we will draw four circles using a loop

```
size(300,300);

for (int i = 1 ; i < 5 ; i += 1) {
    ellipse(i * 50, 100, 30, 30);
    println(i);
}
```

Example. for loop drawing





equivalent code

```
size(300,300);

for (int i = 1 ; i < 5 ; i += 1) {
    ellipse(i * 50, 100, 30, 30);
}
```

```
size(300,300);

ellipse(50,  100, 30, 30); // i is 1
ellipse(100, 100, 30, 30); // i is 2
ellipse(150, 100, 30, 30); // i is 3
ellipse(200, 100, 30, 30); // i is 4
```

Example 1. Drawing with For Loops

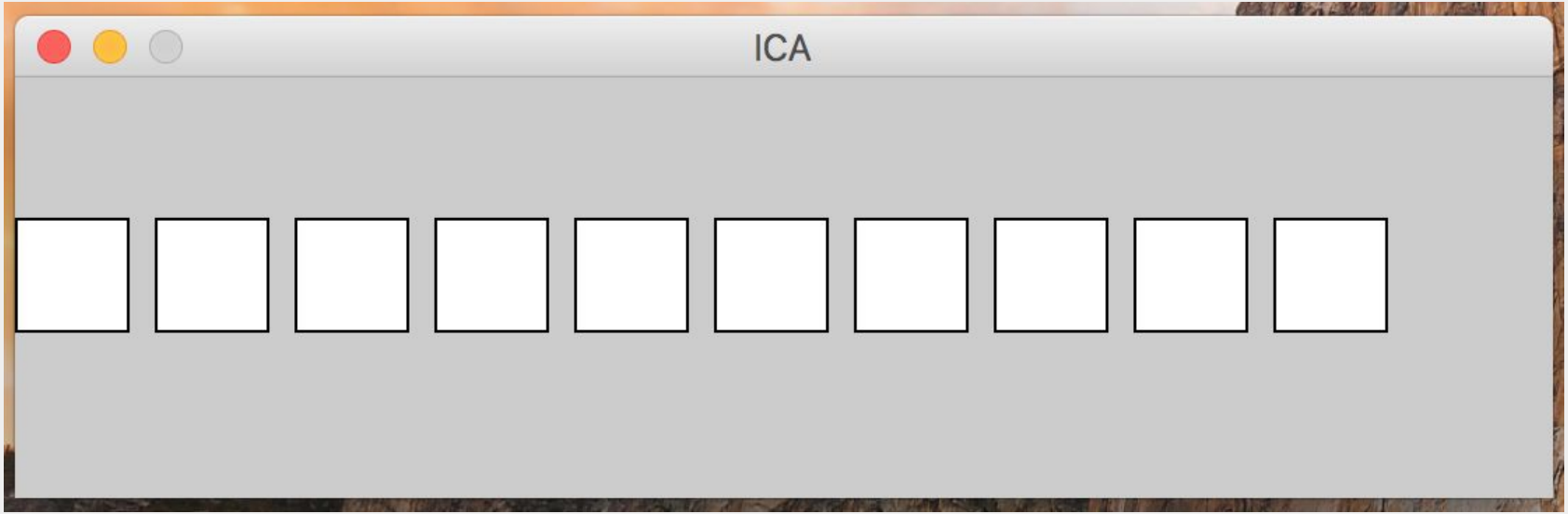
What is the output?

```
size(550, 150);
for (int i = 5 ; i < 550 ; i += 50) {
   rect(i, 50, 40, 40);
 }
```

Drawing it by hand

```
size(550, 150);
for (int i = 5 ; i < 550 ; i += 50) {
   rect(i, 50, 40, 40);
 }
```

Example 1. For Loops - What is the output?

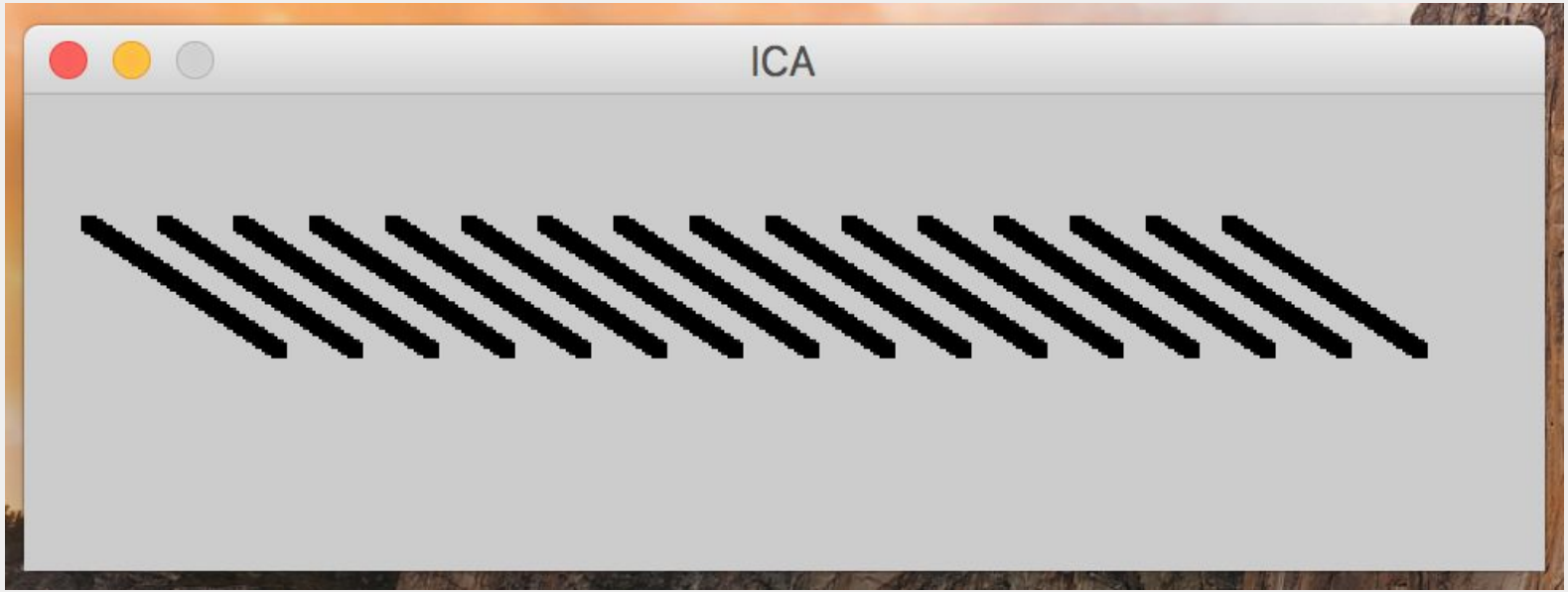


Example 2 - What is the output?

- What kind of canvas will this code produce?
- Again, no computers
- Notice the differences compared to the last example

```
void setup() {
  size(480, 120);
  strokeWeight(5);
}

void draw() {
  for (int i = 20 ; i < 400 ; i += 24) {
    line(i, 40, i+60, 80);
  }
}
```

Example 2 - What is the output?

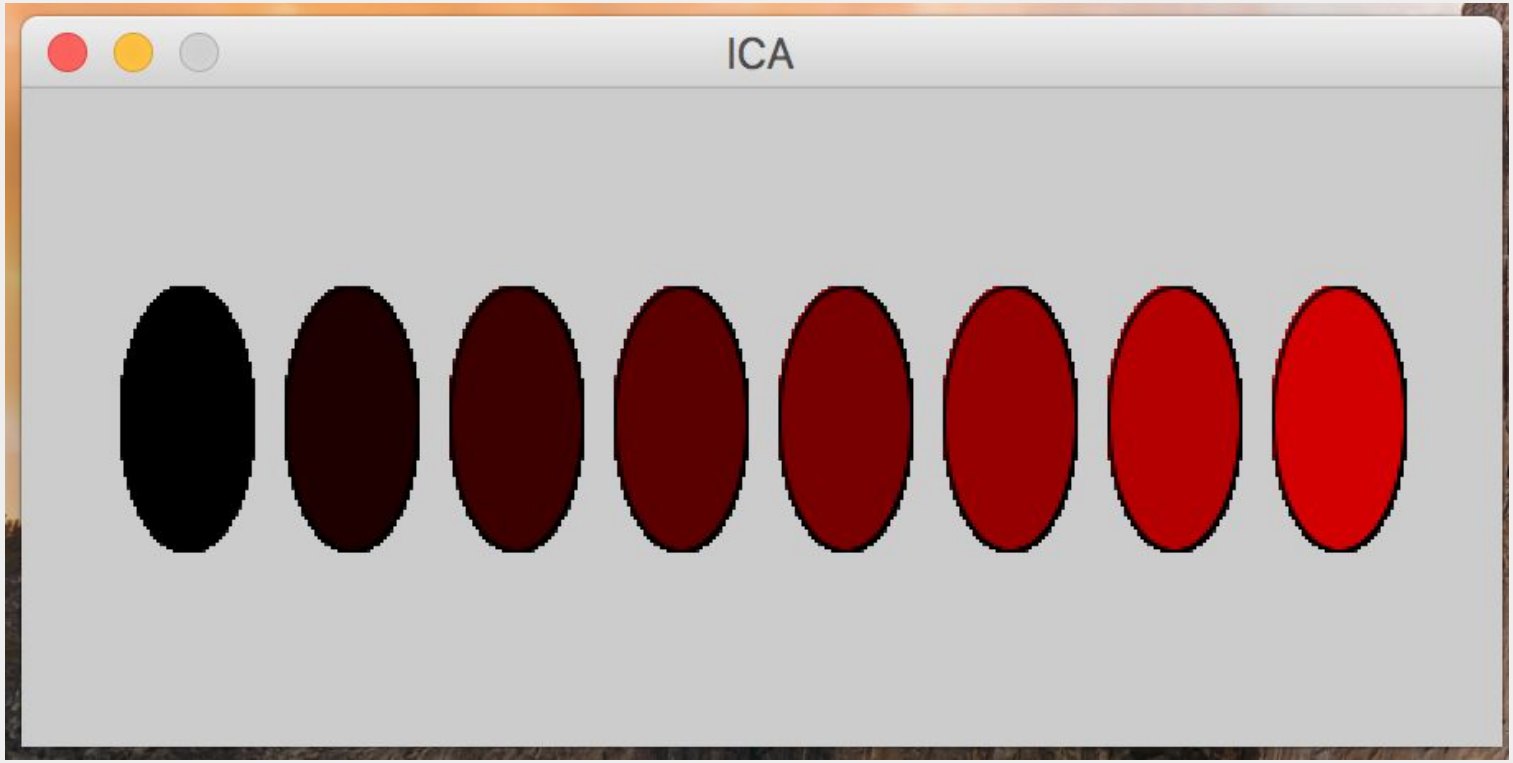


Example 2 - What is the output?

- What kind of canvas will this code produce?
- Notice how we are using the **loop-variable** to determine the location and color of the shapes

```
void setup() {
  size(450, 200);
}

void draw() {
  for (int i = 0 ; i < 8 ; i += 1) {
    fill(i*30, 0, 0);
    ellipse(50 + i * 50, 100, 40, 80);
  }
}
```

Example 3 - What is the output?

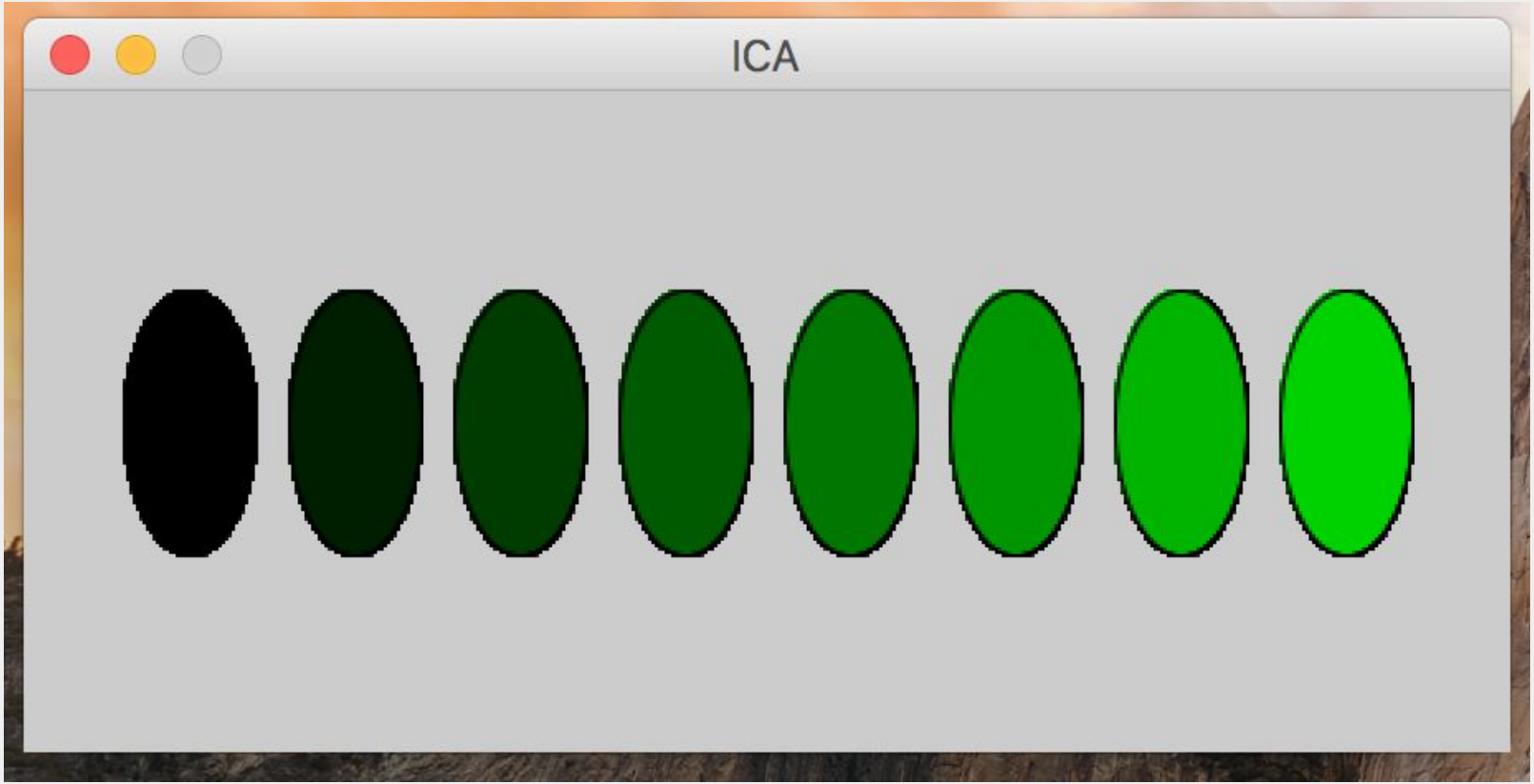


Example 3 - What is the output?

- What kind of canvas will this code produce?

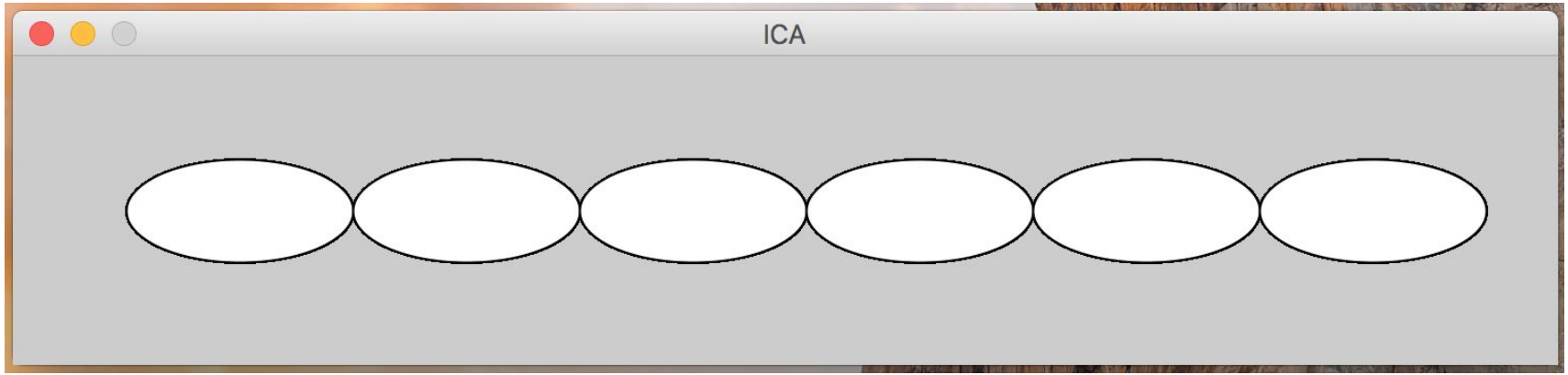
```
void setup() {
  size(450, 200);
}

void draw() {
  for (int i = 0 ; i < 8 ; i += 1) {
    fill(0, i*30, 0);
    ellipse(50 + i * 50, 100, 40, 80);
  }
}
```

Example 4 - What is the output?



Example 5 - Writing a program

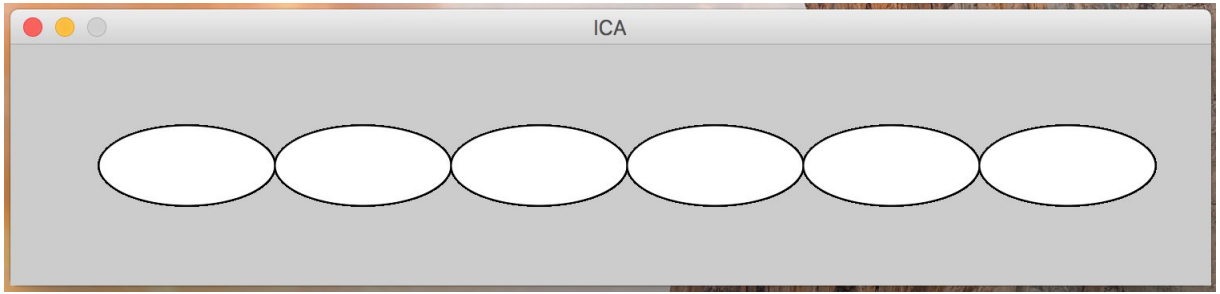
Write a processing program that produces a canvas like this one, using for-loops



Solution - v1

```
void setup() {
  size(750, 150);
}

void draw() {
  for (int i = 1 ; i < 7 ; i += 1) {
    ellipse(i*110, 75, 110, 50);
  }
}
```

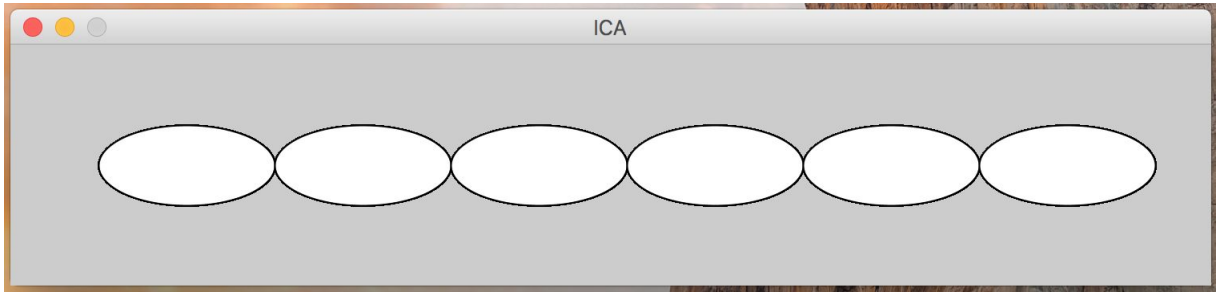
What's another way this could be done?



Solution - v2

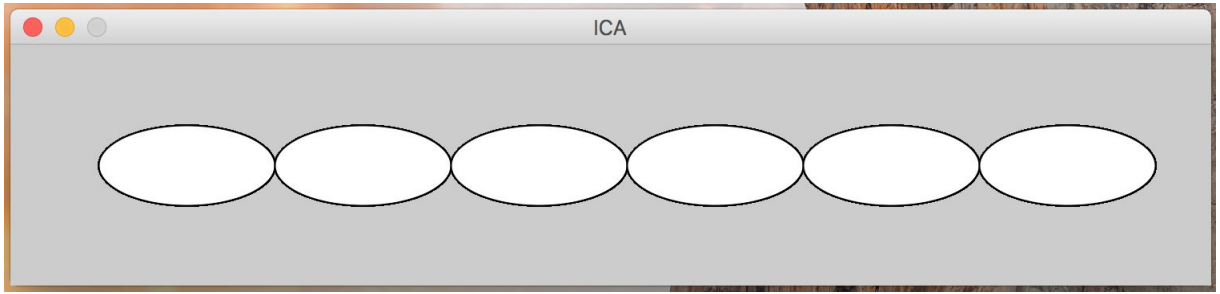
```
void setup() {
  size(750, 150);
}

void draw() {
  for (int i = 110 ; i < 750 ; i += 110) {
    ellipse(i, 75, 110, 50);
  }
}
```

What's another way this could be done?



Solution - v3

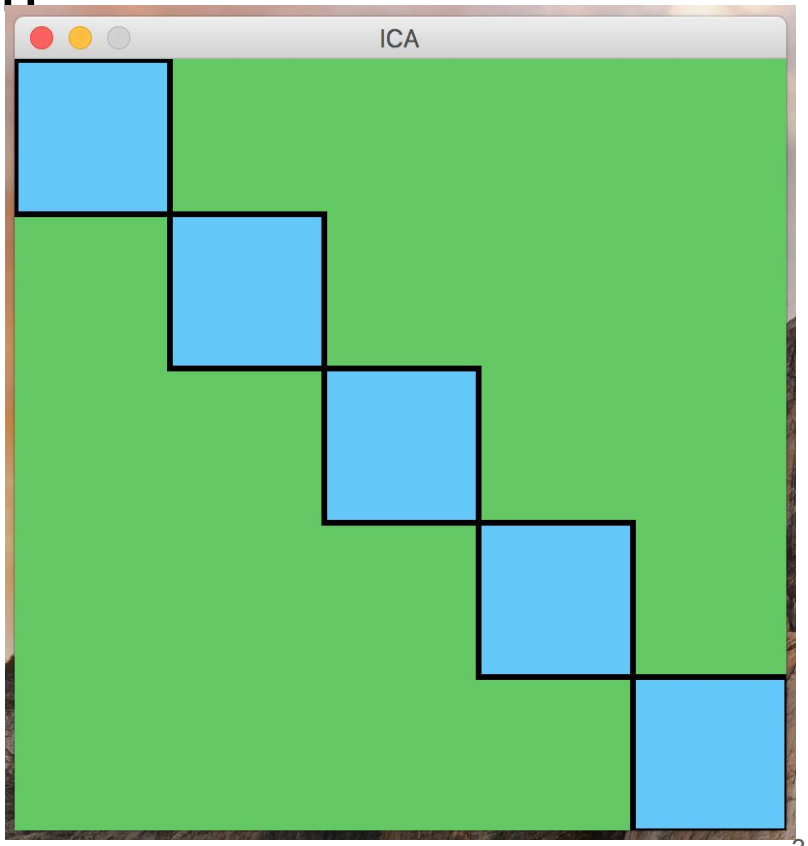
```
void setup() {
  size(750, 150);
}

void draw() {
  for (int i = 650 ; i > 0 ; i -= 110) {
    ellipse(i, 75, 110, 50);
  }
}
```



Example 6 - Writing a program

Write a processing program that produces a canvas like this one

Think: How do the x and y coordinates need to change for each rectangle shown?



Example 6 - Writing a program

```
void setup() {
  size(400, 400);
  strokeWeight(3);
}

void draw() {
  background(100, 200, 100);
  for (int i = 0 ; i < 400 ; i += 80) {
    fill(100, 200, 250);
    rect(i, i, 80, 80);
  }
}
```

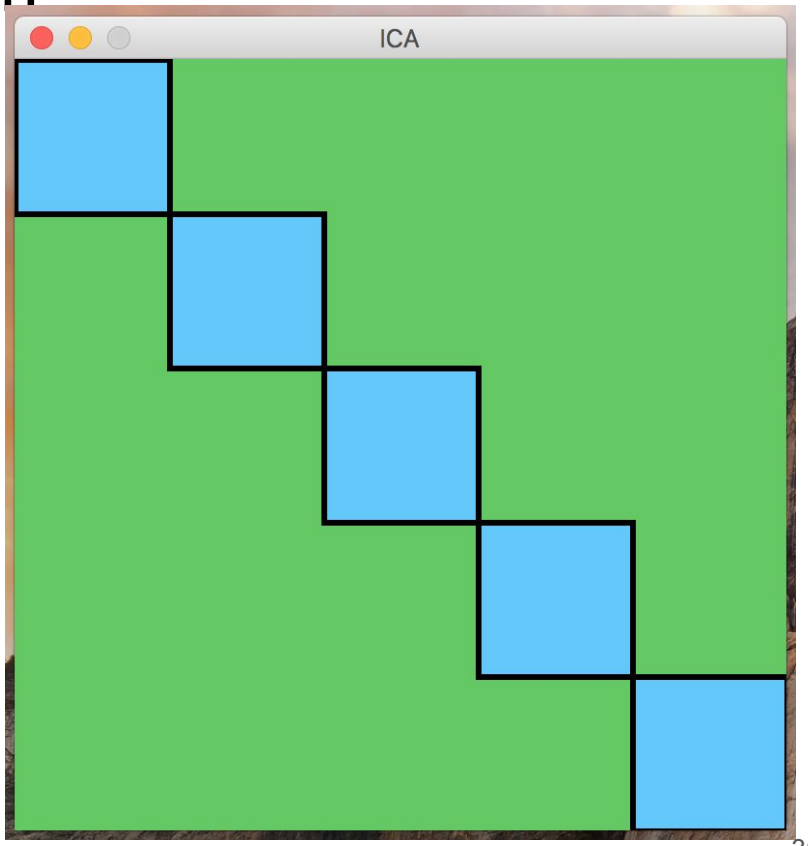


Image for question 1

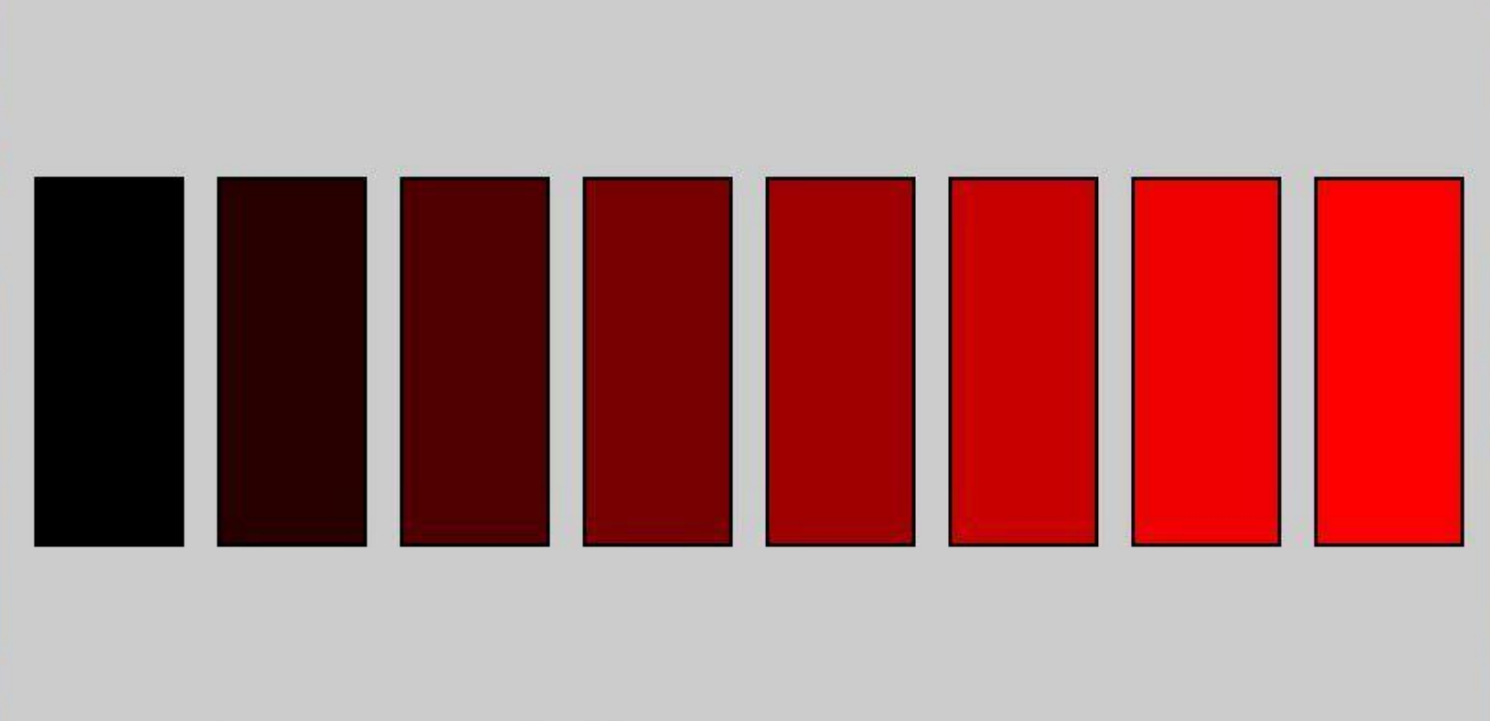


Image for question 2

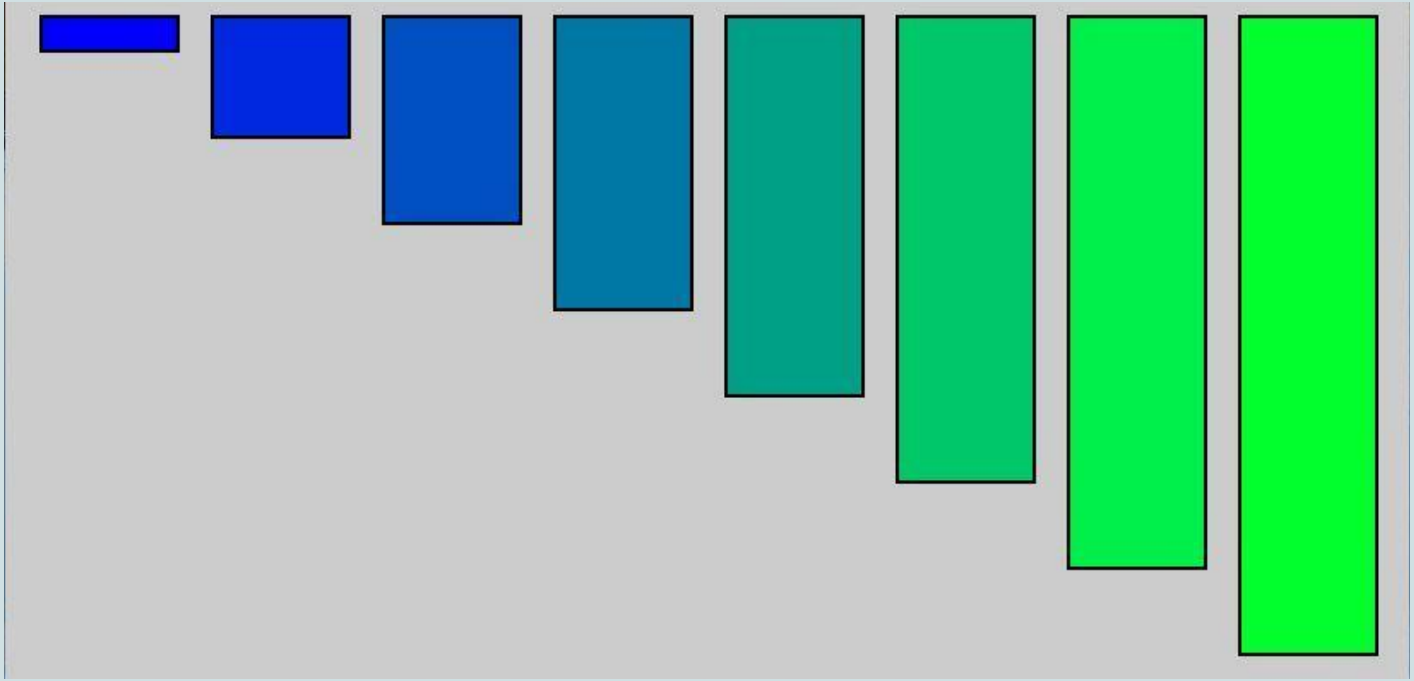


Image for question 3