Which came first?

A) Chicken
B) Egg
C) Both
D) Neither
Figuring out the ORDER in which a computer is commanded to do steps is CRITICAL to programming!

Logic or data dependency: for example,
• Before you can add two numbers, you have to get them into the computer.
• Before you can copy a little digital image into a big EditablePicture, you have to get the little digital image into the computer and then make a Picture from it.
iClicker Suppose you want to fit 6 images into a big EditablePicture so the EditablePicture is just the right width to fit them horizontally. What order of steps is required?
A) 1: make the EditablePicture
   2: input the 6 images
   3: run a loop to sum their widths

B) 1: run a loop to sum their widths
   2: make the EditablePicture
   3: input the 6 images

C) 1: input the 6 images
   2: make the EditablePicture
   3: run a loop to sum their widths

D) 1: input the 6 images
   2: run a loop to sum their widths
   3: make the EditablePicture
Chromakey (Proj04) ordering problem
public class PictureApp {
    public static void main(String[] a) {
       FileChooser.pickMediaPath();
        EditablePicture portrait =
            new EditablePicture(Chooser.pickFile());
        Picture costume =
            new Picture(Chooser.pickFile());
        portrait.substForRedTiled(costume);
        portrait.explore();
        Picture background =
            new Picture(Chooser.pickFile());
        portrait.substForGreenTiled(background);
        portrait.explore();
    }
}
Chromakey (Proj04) ordering problem

Method call:

```
portrait.substForRedTiled(costume);
```

Method call’s effect:

1
Chromakey (Proj04) ordering problem

2nd Method call:
portrait.substForGreenTiled(background);

2nd Method call’s effect:
Input the portrait

Input the costume

Input the background

Change each very red pixel in the portrait so its color comes from the right pixel in the costume

portrait with red substituted

Change each very green pixel in the (changed) portrait so its color comes from the right pixel in the background

BAD RESULT!
Chromakey (Proj04) ordering problem

Try to solve it by doing both substitutions in one method. The INSTRUCTIONS in Proj4 said to do that!

portrait.chromakeyRG(background, costume);
Input portrait

Input costume

Input background

chromakey RG ( , )

Do both substitutions together!
for (int x = 0; x < this.get_width(); x++)
    for (int y = 0; y < this.get_height(); y++)
        if (my_chrome.is_very_red())
            my_chrome.set_color(same_color_for_chrome);
        if (my_chrome.is_very_green())
            my_chrome.set_color(same_color_for_background);
public void chromakeyRG(Picture forGreen, Picture forRed)
{
    for( int x=0; x<this.getWidth(); x++)
    {
        for( int y=0; y<this.getHeight(); y++)
        {
            Pixel maybeChange = this.getPixel(x,y);
            if( maybeChange.getRed()>
                maybeChange.getGreen()+maybeChange.getBlue() )
            {
                int xsub, ysub;
                xsub = (x % forRed.getWidth());
                ysub = (y % forRed.getHeight());
                Pixel fromsubs = forRed.getPixel( xsub, ysub );
                maybeChange.setColor( fromsubs.getColor() );
            }
            if( maybeChange.getGreen()>
                maybeChange.getRed()+maybeChange.getBlue() )
            {
                int xsub, ysub;
                xsub = (x % forGreen.getWidth());
                ysub = (y % forGreen.getHeight());
                Pixel fromsubs = forGreen.getPixel( xsub, ysub );
                maybeChange.setColor( fromsubs.getColor() );
            }
        }
    }
}
Warning when you read this for review: It's STILL WRONG.
Pixel mayChange = this.getPixel(x, y)

if (mayChange is very red)
    Yes! Red!
else
    no

if (mayChange is very green)
    Yes! Green!
else
    no, not green

mayChange.setColor (color from costume)

mayChange.setColor (color from background)
public void chromakeyRG(Picture forGreen, Picture forRed) {
    for( int x=0; x<this.getWidth(); x++) {
        for( int y=0; y<this.getHeight(); y++) {
            Pixel maybeChange = this.getPixel(x,y);
            if( maybeChange.getRed() > maybeChange.getGreen()+maybeChange.getBlue() ) {
                int xsub, ysub;
                xsub = (x % forRed.getWidth());
                ysub = (y % forRed.getHeight());
                Pixel fromsubs = forRed.getPixel( xsub, ysub );
                maybeChange.setColor( fromsubs.getColor() );
            }
            if( maybeChange.getGreen() > maybeChange.getRed()+maybeChange.getBlue() ) {
                int xsub, ysub;
                xsub = (x % forGreen.getWidth());
                ysub = (y % forGreen.getHeight());
                Pixel fromsubs = forGreen.getPixel( xsub, ysub );
                maybeChange.setColor( fromsubs.getColor() );
            }
        } //end of inner for
    } //end of outer for
} //end of body of chromakeyRG
Pixel mayChange = this.getPixel(x,y);

if (mayChange is very red) yes! Red

no (maybe it's green?)

if (mayChange is very green) yes! Green

mayChange.setColor(color from background)

Correct flowchart for processing one pixel.
public void chromakeyRG(Picture forGreen, Picture forRed) {
    for( int x=0; x<this.getWidth(); x++) {
        for( int y=0; y<this.getHeight(); y++) {
            Pixel maybeChange = this.getPixel(x,y);
            if( maybeChange.getRed() > maybeChange.getGreen() + maybeChange.getBlue() ) {
                int xsub, ysub;
                xsub = (x % forRed.getWidth());
                ysub = (y % forRed.getHeight());
                Pixel fromsubs = forRed.getPixel( xsub, ysub );
                maybeChange.setColor( fromsubs.getColor() );
            }
            else {
                if( maybeChange.getGreen() > maybeChange.getRed()+maybeChange.getBlue() ) {
                    int xsub, ysub;
                    xsub = (x % forGreen.getWidth());
                    ysub = (y % forGreen.getHeight());
                    Pixel fromsubs = forGreen.getPixel( xsub, ysub );
                    maybeChange.setColor( fromsubs.getColor() );
                } // end of if very green
            } // end of else
        } // end of inner for
    } // end of outer for
} // end of method body
public void chromakeyRG(Picture forGreen, Picture forRed)
{
    for (int x=0; x<this.getWidth(); x++)
    {
        for (int y=0; y<this.getHeight(); y++)
        {
            Pixel maybeChange = this.getPixel(x,y);
            if( maybeChange.getRed() > maybeChange.getGreen() + maybeChange.getBlue() )
            {
                int xsub, ysub;
                xsub = (x % forRed.getWidth());
                ysub = (y % forRed.getHeight());
                Pixel fromsubs = forRed.getPixel( xsub, ysub );
                maybeChange.setColor( fromsubs.getColor() );
            } //end of if very red
            else
            {
                if( maybeChange.getGreen() > maybeChange.getRed()+maybeChange.getBlue() )
                {
                    int xsub, ysub;
                    xsub = (x % forGreen.getWidth());
                    ysub = (y % forGreen.getHeight());
                    Pixel fromsubs = forGreen.getPixel( xsub, ysub );
                    maybeChange.setColor( fromsubs.getColor() );
                } //end of if very green
            } //end of else
        } //end of inner for
    } //end of outer for
} //end of method body
Pixel mayChange = this.getPixel(x,y);

if (mayChange is very red)
    yes! Red

if (mayChange is very green)
    yes! Green

MayChange. set Color (color from costume)

MayChange. set Color (color from background)