Computer Programming Fundamentals

CS 152
Professor: Leah Buechley
TAs: Melody Horn, Noah Garcia, Andrew Geyko, Juan Ormaza
Time: MWF 10:00-10:50am
https://handandmachine.cs.unm.edu/classes/CS152_Fall2021/

MIDTERM GRADES POSTED

PARTICIPATION: ATTENDENCE

questions?

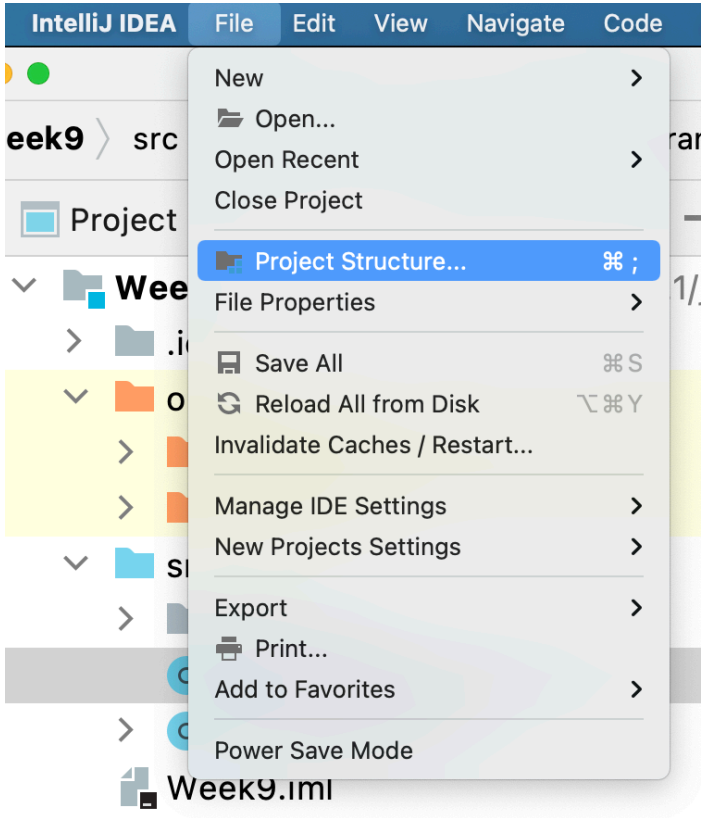
OPEN UP PROJECT FROM LAST CLASS

```
public class DrawingApplication extends JPanel implements KeyListener, MouseMotionListener, MouseListener {
    int width;
    int height;
    char keyPressed;
    int keyCode;
    int mouseX, mouseY;
    Graphics g;
    Color drawingColor;
    //color picker variables
    final int squareSize = 50;
    final int spacing = 20;
    final Color[] colors = {new Color(238, 82, 101),
            new Color(238, 147, 82),
            new Color(233, 203, 27),
            new Color(119, 187, 95),
            new Color(82, 163, 238),
            new Color(134, 82, 238),
            new Color(0, 0, 0)};

    DrawingApplication() {
        width = 600;
        height = 510;
        drawingColor = Color.BLACK;
        Dimension d = new Dimension(width, height);
        setPreferredSize(d);
        addKeyListener(this);
        addMouseMotionListener(this);
        addMouseListener(this);
        setFocusable(true);
        requestFocusInWindow();
        setVisible(true);
        System.out.println("The initial value of mouseX is:" + mouseX);
        System.out.println("The initial value of mouseY is:" + mouseY);
    }

    public static void main(String[] args) {
        DrawingApplication panel = new DrawingApplication();
        MyFrame f = new MyFrame(panel);
        //panel.animate(60);
    }

    @Override
    protected void paintComponent(Graphics g) {
        super.paintComponent(g);
        setBackground(Color.WHITE);
        //color picker variables
        int x,y;
        x = width - squareSize - spacing;
        y = spacing;
        for (int i=0;i<colors.length;i++) {
            g.setColor(colors[i]);
            g.fillRect(x, y, squareSize,squareSize);
            y = y + squareSize + spacing;
        }
    }

    void delay(int time) {
        try {
            Thread.sleep(time);
        } catch (Exception exc) {
        }
    }

    void animate(int framerate) {
        int delay = 1000 / framerate;
        while (true) {
            repaint();
            delay(delay);
        }
    }

    @Override
    public void keyTyped(KeyEvent e) {
        keyPressed = e.getKeyChar();
        System.out.println(keyPressed);
        //clear screen if spacebar is pressed
        if (keyPressed == ' ') {
            repaint();
        }
    }
```

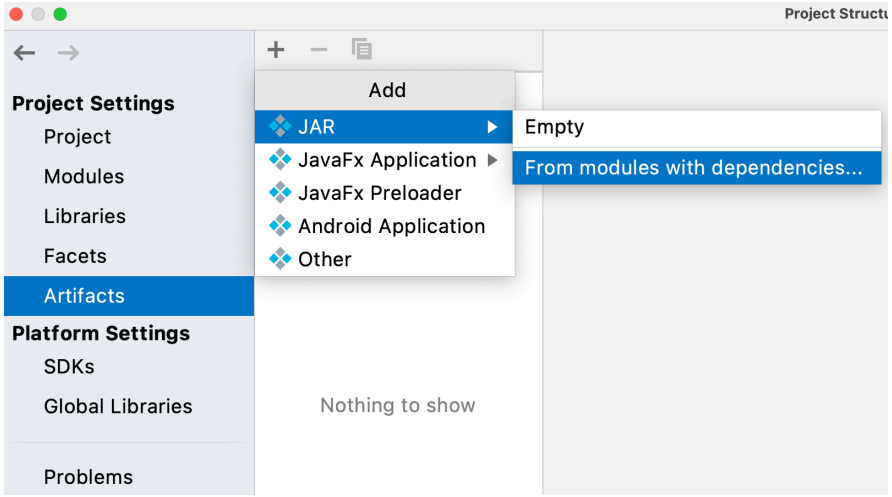
COMPLETE CODE IS POSTED ONLINE

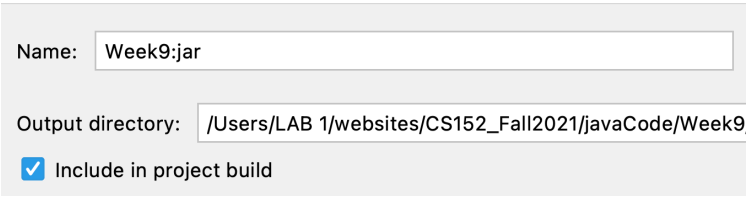
EXPORT AN APPLICATION

OPEN PROJECT STRUCTURE



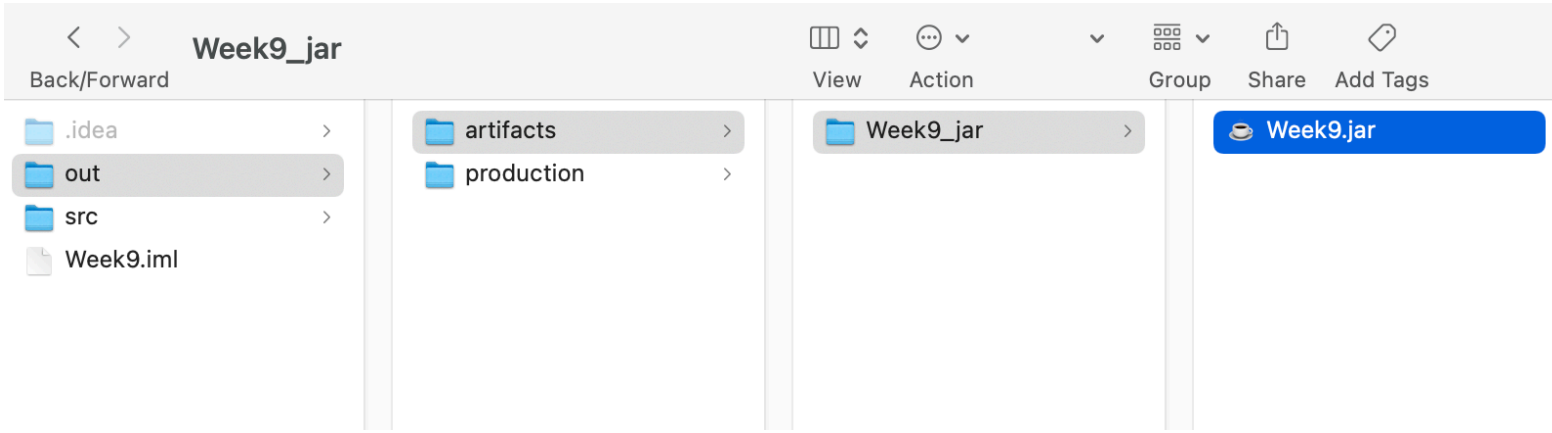
IN PROJECT STRUCTURE





- Under “Artifacts” click +
- Choose JAR —> From modules with dependencies…
- Choose “DrawingApplication” as Main Class
- Check “Include in project build”

BROWSE TO PROJECT FOLDER ON COMPUTER



- Double click on Week9.jar file to run your application

questions?

NEW TOPIC: RECURSION

RECURSION

defining a problem in terms of itself

AN EXAMPLE: FACTORIAL

Factorial

From Wikipedia, the free encyclopedia

In mathematics, the **factorial** of a positive integer n, denoted by $n!$, is the product of all positive integers less than or equal to n:

$$n! = n \cdot (n-1) \cdot (n-2) \cdot (n-3) \cdot \cdots \cdot 3 \cdot 2 \cdot 1 \,.$$

For example,

$$5! = 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 120 \,.$$

```
x! = x * (x-1) * (x-2) * (x-3) … * 2 * 1
```

$$5! = 5 * 4 * 3 * 2 * 1$$

$$5! = 120$$

RECURSION

defining a problem in terms of itself

```
5! = 5 * 4 * 3 * 2 * 1
```

```
5! = 5 * 4!
4! = 4 * 3!
3! = 3 * 2!
2! = 2 * 1!
1! = 1
```

factorial in code

CREATE A NEW PROJECT
“Week10”

CREATE A NEW CLASS
“Factorial”

A FACTORIAL PROGRAM

```
public class Factorial {

    public static void main(String[] args) {

    }

    static int factorial (int x) {

   }
}
```

A FACTORIAL PROGRAM

```
public class Factorial {

    public static void main(String[] args) {

    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;

       return result;
    }
}
```

A FACTORIAL PROGRAM

```
public class Factorial {

    public static void main(String[] args) {

    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

recursion!

A FACTORIAL PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
```

LET’S SEE HOW THIS WORKS

“THE STACK”

a list of things the computer needs to do

“THE STACK”
“THE CALL STACK”
“THE STACK FRAME”

RUNNING THE PROGRAM

STACK
list of things computer needs to do

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

factorial(5);

“push” = add something to stack

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
        factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
result = 2 * factorial(1);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
        factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
result = 2 * factorial(1);
        factorial(1);
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

STACK
list of things computer needs to do

```
        factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
result = 2 * factorial(1);
         result = 1
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
```

STACK
list of things computer needs to do

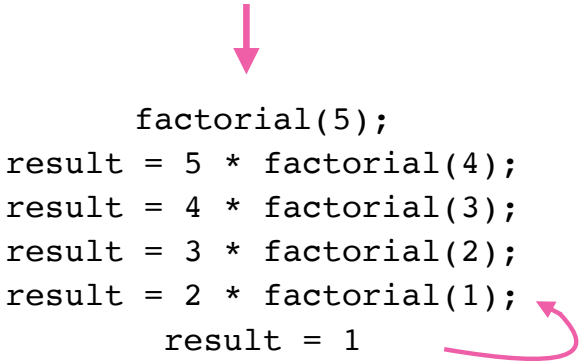
```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
result = 2 * factorial(1);
result = 1
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
```

STACK
list of things computer needs to do



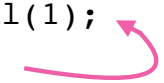
RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
```

STACK
list of things computer needs to do

```
      factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
result = 2 * factorial(1);
        result = 1
```



“pop” = remove something from stack

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
```

STACK
list of things computer needs to do

```
      factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
     result = 2 * 1;
```

“pop” = remove something from stack

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
```

STACK
list of things computer needs to do

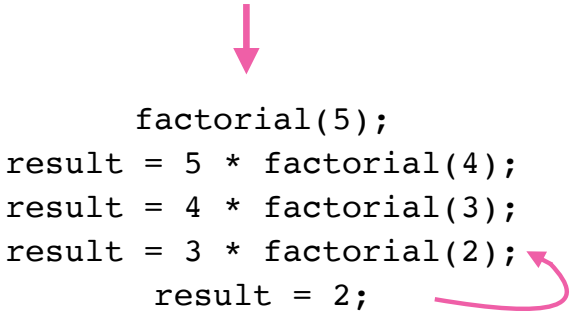
```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * factorial(2);
result = 2;
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
```

STACK
list of things computer needs to do



RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 3 * 2;
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
```

STACK
list of things computer needs to do

```
        factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
        result = 6;
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 4 * factorial(3);
result = 6;
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 4 * 6;
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * factorial(4);
result = 24;
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
```

STACK
list of things computer needs to do


```
        factorial(5);
result = 5 * factorial(4);
        result = 24;
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
```

STACK
list of things computer needs to do

```
factorial(5);
result = 5 * 24;
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
```

STACK
list of things computer needs to do

```
factorial(5);
result = 120;
```

RUNNING THE PROGRAM

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
```

STACK
list of things computer needs to do

```
factorial(5);
result = 120;
```

RUNNING THE PROGRAM

STACK
list of things computer needs to do

120

```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
```

RUNNING THE PROGRAM

STACK
list of things computer needs to do

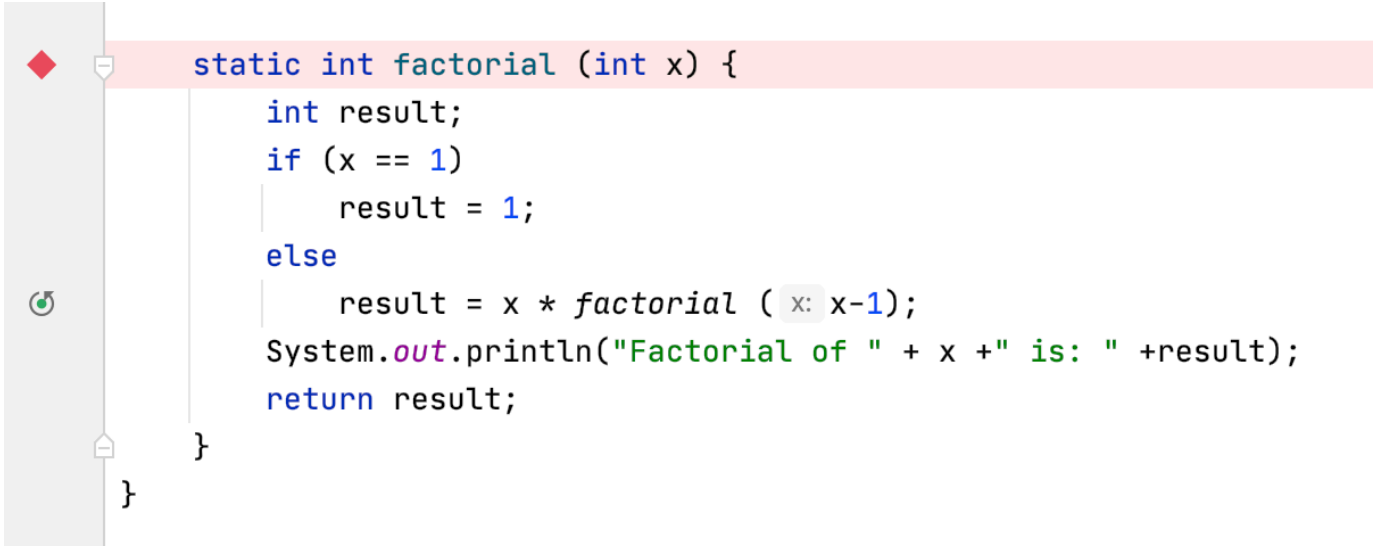
```
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
```

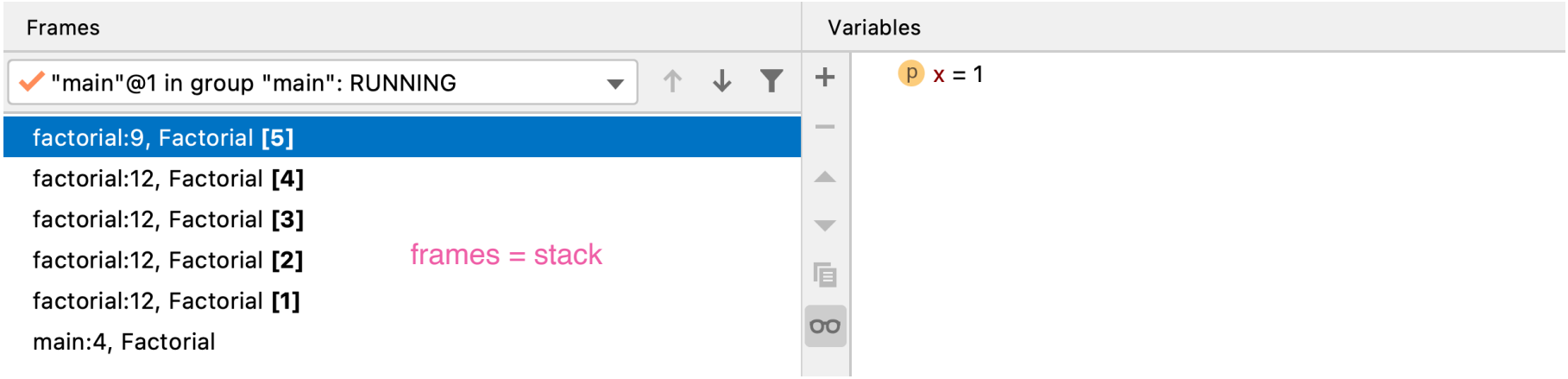
questions?

LET’S RUN IN THE DEBUGGER

ADD A BREAKPOINT BY factorial()



DEBUGGER


Frames
Variables
"main"@1 in group "main": RUNNING
factorial:9, Factorial [5]
factorial:12, Factorial [4]
factorial:12, Factorial [3]
factorial:12, Factorial [2]
factorial:12, Factorial [1]
main:4, Factorial
frames = stack
p x = 1

questions?

RUNNING THE PROGRAM

```java
public class Factorial {

    public static void main(String[] args) {
        factorial(5);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
```

RUNNING THE PROGRAM

```java
public class Factorial {

    public static void main(String[] args) {
        factorial(20);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
Factorial of 6 is: 720
Factorial of 7 is: 5040
Factorial of 8 is: 40320
Factorial of 9 is: 362880
Factorial of 10 is: 3628800
Factorial of 11 is: 39916800
Factorial of 12 is: 479001600
Factorial of 13 is: 1932053504
Factorial of 14 is: 1278945280
Factorial of 15 is: 2004310016
Factorial of 16 is: 2004189184
Factorial of 17 is: -288522240
Factorial of 18 is: -898433024
Factorial of 19 is: 109641728
Factorial of 20 is: -2102132736
```

WHAT'S GOING ON HERE??

```java
public class Factorial {

    public static void main(String[] args) {
        factorial(20);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
Factorial of 6 is: 720
Factorial of 7 is: 5040
Factorial of 8 is: 40320
Factorial of 9 is: 362880
Factorial of 10 is: 3628800
Factorial of 11 is: 39916800
Factorial of 12 is: 479001600
Factorial of 13 is: 1932053504
Factorial of 14 is: 1278945280
Factorial of 15 is: 2004310016
Factorial of 16 is: 2004189184
Factorial of 17 is: -288522240
Factorial of 18 is: -898433024
Factorial of 19 is: 109641728
Factorial of 20 is: -2102132736
```

ideas?

BASIC NUMBER TYPES IN JAVA

TYPE	# BITS	minimum value	maximum value	example
byte	8	-128	127	53
int	32	-2,147,483,648	2,147,483,647	3079
float	32	~ -3.4×10^{38} with 7 significant digits	~ 3.4×10^{38} with 7 significant digits	4.589

NUMBERS TOO BIG FOR MEMORY ALLOCATED BY INT

```java
public class Factorial {

    public static void main(String[] args) {
        factorial(20);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
Factorial of 6 is: 720
Factorial of 7 is: 5040
Factorial of 8 is: 40320
Factorial of 9 is: 362880
Factorial of 10 is: 3628800
Factorial of 11 is: 39916800
Factorial of 12 is: 479001600
Factorial of 13 is: 1932053504
Factorial of 14 is: 1278945280
Factorial of 15 is: 2004310016
Factorial of 16 is: 2004189184
Factorial of 17 is: -288522240
Factorial of 18 is: -898433024
Factorial of 19 is: 109641728
Factorial of 20 is: -2102132736
```

NEED MORE SPACE THAN 32 BITS OF INT

```java
public class Factorial {

    public static void main(String[] args) {
        factorial(20);
    }

    static int factorial (int x) {
        int result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
Factorial of 6 is: 720
Factorial of 7 is: 5040
Factorial of 8 is: 40320
Factorial of 9 is: 362880
Factorial of 10 is: 3628800
Factorial of 11 is: 39916800
Factorial of 12 is: 479001600
Factorial of 13 is: 1932053504
Factorial of 14 is: 1278945280
Factorial of 15 is: 2004310016
Factorial of 16 is: 2004189184
Factorial of 17 is: -288522240
Factorial of 18 is: -898433024
Factorial of 19 is: 109641728
Factorial of 20 is: -2102132736
```

MORE NUMBER TYPES IN JAVA

TYPE	# BITS	minimum value	maximum value	example
short	16	-32,768	32,767	134
long	64	~ -9.2×10^{18}	~ 9.2×10^{18}	30,790
double	64	~ -1.7×10^{308} with 15 significant digits	~ 1.7×10^{308} with 15 significant digits	10,789.998

A FIX (THAT WILL ALSO EVENTUALLY FAIL)

```java
public class Factorial {

    public static void main(String[] args) {
        factorial(20);
    }

    static int factorial (int x) {
        long result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 1 is: 1
Factorial of 2 is: 2
Factorial of 3 is: 6
Factorial of 4 is: 24
Factorial of 5 is: 120
Factorial of 6 is: 720
Factorial of 7 is: 5040
Factorial of 8 is: 40320
Factorial of 9 is: 362880
Factorial of 10 is: 3628800
Factorial of 11 is: 39916800
Factorial of 12 is: 479001600
Factorial of 13 is: 6227020800
Factorial of 14 is: 87178291200
Factorial of 15 is: 1307674368000
Factorial of 16 is: 20922789888000
Factorial of 17 is: 355687428096000
Factorial of 18 is: 6402373705728000
Factorial of 19 is: 121645100408832000
Factorial of 20 is: 2432902008176640000
```

THE FAIL

```
public class Factorial {

    public static void main(String[] args) {
        factorial(21);
    }

    static int factorial (int x) {
        long result;
        if (x == 1)
            result = 1;
        else
            result = x * factorial(x-1);
        System.out.println("Factorial of " + x +" is: " + result);
        return result;
    }
}
```

```
Factorial of 20 is: 2432902008176640000
Factorial of 21 is: -4249290049419214848
```

questions?

GOOD TO KNOW THESE THINGS ARE POSSIBLE

WE ARE LIMITED BY THE PHYSICAL CONSTRAINTS OF COMPUTERS

WHAT WE CAN DO IS LIMITED BY SPACE AND TIME

THEORETICAL COMPUTER SCIENCE: UNDERSTANDING THESE LIMITS