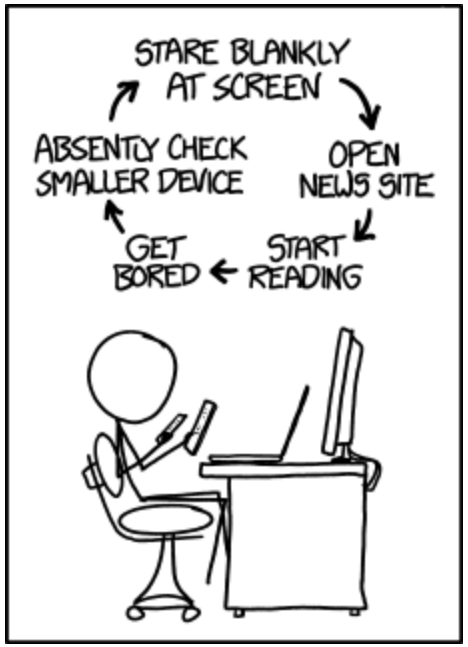
Mr. Fahrenbacher
AP Computer Science
Unit 3B - Definite and Indefinite Loops

List of Java Concepts and Vocabulary From This Unit

- For Loops
- Math Sequences
- Accumulation
- Scope
- Finding Max Value
- While Loops
- For-While Loop Equivalence
- Infinite Loops



Definite Loops

Read the program below and compare the code with the output. Try to deduce how the new code affects how the program runs. Why do you think the lopps are called **Definite Loops**?

```
public class CountUp {

	public static void main(String[] args) {

		int i;
		for(i = 1; i <= 4; i++) {
			System.out.print(i + ", ");
		}
		System.out.println(i);
	}
}

Program Output: 1, 2, 3, 4, 5
```

```
public class CountDown {

	public static void main(String[] args) {

		for(int num = 10; num > 5; num = num - 1) {
			System.out.print(num + ", ");
		}
		//System.out.println(num); <= illegal!
	}
}

Program Output: 10, 9, 8, 7, 6,
```

```
for(;;) {
	System.out.print("A");
}
```

The Life Cycle of the For Loop (1-2-3-4-5-3-4-5-3-4-5-3-4-5-3-6)

1 Create variables / Other Stuff / PRE LOOP

for(2 INITIALIZE A VARIABLE**;** 3 BOOLEAN EXPR KEEP GOING CONDITION!**;** 5 VARIABLE MODIFICATION) {

4 BODY OF LOOP - WHAT YOU WANT TO REPEAT

}

6 POST LOOP CODE (AFTER Step 3 is false)

Loops and Drawing (http://semmy.me/ide/)

Below is code for another language called Processing, which is very similar to Java. Copy this code into this online ide for Processing. Can you modify the code to use a loop to draw the 8 ellipses?

```
void setup()
{
    //sets the size of the drawing area
    size(400,400);
}

void draw(){

    //filles the background with the color white
    background(255, 255, 255);

    //sets the fill color to green (r, g, b) - each between 0 and 255
    fill(100, 200, 50);

    //draws 8 ellipses - (x, y, width, height)
    ellipse(100, 100, 20, 20);
    ellipse(100, 120, 20, 20);
    ellipse(100, 140, 20, 20);
    ellipse(100, 160, 20, 20);
    ellipse(100, 180, 20, 20);
    ellipse(100, 200, 20, 20);
    ellipse(100, 220, 20, 20);
    ellipse(100, 240, 20, 20);
}
```

Pattern in Words

Initial y: 100
How to get to the next y: 20
How many ellipses? 8

Pattern In a Table

Ellipse Number (i)	1	2	3	4	5	6	7	8	i
Y-Coordinate (y)	100	120	140	160	180	200	220	240	20i + 80

When Does The Loop Actually End? (http://semmy.me/ide/)

Below is code for another language called Processing, which is very similar to Java. Copy this code into this online ide for Processing. How many ellipses do you think the code below will draw?

```
void setup()
{
    //sets the size of the drawing area
    size(400,400);
}

void draw(){

    //filles the background with the color white
    background(255, 255, 255);

    //sets the fill color to green (r, g, b) - each between 0 and 255
    fill(100, 200, 50);

    //draws ??? ellipses
    for(int i = 1; i <= 5; i++) {
          i++;
          ellipse(100, 20*i, 20, 20);
     }
}
```

Math Sequences

Below is a program that demonstrates two different strategies for a for loop to generate the first _____ perfect squares. Compare and contrast the two strategies - pay close attention to the use of the variable i.

```
public class PerfectSquares {

	public static void main(String[] args) {

		//Approach #1
		for(int i = 0; i <= 10; i++) {
			System.out.print(i * i + " ");
		}

		System.out.println();

		//Approach #2
		int perfectSquare = 0;


		for(int i = 0; i <= 10; i++) {
			int diff = 1;
			System.out.print(perfectSquare + " ");

			perfectSquare += diff;
			diff += 2;
		}
	}
}
```

Program Output: **0 1 4 9 16 25 36 49 64 81 100**
0 1 4 9 16 25 36 49 64 81 100

Accumulating a Sequence (Series)

Consider the program below - what is the code trying to do? How does it accomplish its intended task?

```
public class TriangleNumber {

	public static void main(String[] args) {

		int sum = 0; //accumulator (preloop)
		for(int i = 0; i <= 100; i++) {
			sum += i;   //accumulation
		}

		System.out.println("The sum of integers from 0 to 100 is " + sum);
	}
}
```

Accumulating Values From a User

Consider the program below - what is the code trying to do? How does it accomplish its intended task?

```
import java.util.Scanner;

public class TotalBill {

	public static void main(String[] args) {

		Scanner input = new Scanner(System.in);

		int sum = 0;
		for(int i = 0; i < 5; i++) {
			System.out.print("Enter the cost of the next item: ");
			int next = input.nextInt();
			sum += next;
		}

		System.out.println("Your total bill is  " + sum);
	}

}
```

Scope

The previous program is modified below... but this modification creates an error on the print statement. Why do you think it creates an error?

```
import java.util.Scanner;

public class TotalBill {

	public static void main(String[] args) {

		Scanner input = new Scanner(System.in);

		for(int i = 0; i < 5; i++) {
			int sum = 0;
			System.out.print("Enter the cost of the next item: ");
			int next = input.nextInt();
			sum += next;
		}

		//variables DECLARED in a loop/if cease to exist after the loop/if
		System.out.println("Your total bill is  " + sum);
	}

}
```

An If and a For Enter a Restaurant...

Consider the code segments below. Do each of them have the same output?

`for(int num = 10; num > 5; num--) {`
`  System.out.print(num + ", ");`
`}`
`System.out.print(5);`
	`for(int num = 10; num >= 5; num--) {`
	`  System.out.print(num);`
	`  if(num != 5) {`
	`  System.out.print(", ");`
	`  }`
	`}`
		`for(int num = 10; num >= 5; num--) {`
		`  if(num != 10) {`
		`  System.out.println(", ");`
		`  }`
		`  System.out.print(num);`
		`}`

10, 9, 8, 7, 6, 5

Counting Strategy

The program below uses a common strategy to determine if a set of values has a certain property. Identify the strategy.

```
import java.util.Scanner;

public class AllTall {

	public static void main(String[] args) {

		Scanner input = new Scanner(System.in);
		int count = 0;

		for(int i = 0; i < 24; i++) {
			System.out.print("Enter the next student's height in inches: ");
			int height = input.nextInt();

			if(height < 66) {
				count++;
			}
		}

		if(count > 0) {
			System.out.println("We have " + count + " many short students");
		}
		else {
			System.out.println("Everyone is tall!");
		}
	}
}
```

Finding the Max

Read the program below, a modification on an early program we looked at. How does the code go about accomplishing its task. Does it work?

```
public class TheTallest {

	public static void main(String[] args) {

		Scanner input = new Scanner(System.in);
		System.out.print("Enter the first height: ");
		int max = input.nextInt();

		for(int i = 1; i <= 24; i++) {
			System.out.print("Enter the next height: ");
			int next = input.nextInt();

			if(next > max) {
				max = next;
			}
		}

		System.out.println("The tallest height is " + max);
	}
}
```

Indefinite Loops

The code below introduces a new type of loop. How do you think this code works? How is it similar / different than a for loop?

```
int num = 1;

while(num <= 100) {
      num *= 2;
}

System.out.println("First power of 2 greater than 100 is " + num);


int x = 99;
int div = 2;

while(x % div != 0) {
      div++;
}
```

System.out.println("The smallest divisor greater than 1 is " + div);

The Life Cycle of the While Loop

preloop code (some variable that will **control** the loop)

while(**boolean condition that tells us if the loop should continue**) {

code that we want to repeat (body)

code that modifies the control variable (hopefully causing the loop to end at some point!)

}

post loop code

For-While Equality

Can anything done by a while loop also be done by a for loop (or vice-a-versa)? Try this task below.

While	For

`int x = 1;`
`while(x < 10){`
`  System.out.println("A");`
`  x+=2;`
`}`
	`for(int x = 1; x < 10; x+=2) {`
	`  System.out.println("A");`
	`}`

`int sum = 0;`
`int y = 3;`
`while(y > -4) {`
`  sum += y;`
`  y = y- 1;`
`}`
	`int sum = 0;`
	`for(int y = 3; y > -4; y = y - 1){`
	`  sum += y;`
	`}`

Will Someone Make It Stop!

Why is the code below very appropriate for the song?

```
int x = 1;
while(x < 10) {
        System.out.println("This is the song that never ends");
        System.out.println("It just goes on and on my friend");
        System.out.println("Some people started singing it not knowing what it was,");
        System.out.println("And they'll continue singing it forever just because");
}
```

Accumulating An Indefinite Number of Values From a User

Consider the program below - what is the code trying to do? How does it accomplish its intended task?

```
import java.util.Scanner;

public class TotalBill {

	public static void main(String[] args) {

		Scanner input = new Scanner(System.in);

		int sum = 0;
		System.out.print("Enter the cost of the next item, 0 or less to stop: ");
		int next = input.nextInt();
		int count = 0;

		while(next > 0) {

			sum += next;
			count++;

			System.out.print("Enter the cost of the next item, 0 or less to stop: ");
			next = input.nextInt();
		}

		System.out.println("Your total bill is  " + sum + " , with each item averaging " + sum / count);
	}

}
```