**For, Do-While, and Nested Loops**

Reference: Gaddis 4.7, 4.4-4.5

**For Loops**

Ideal when you know before the loop begins

*Example 1: Grade Average*

Maybe you know you want to average 10 test grades. Our while loop was:

final int NUM\_GRADES =

int count =

while(

{

System.out.println(“Please enter the test grade”);

sum += scan.nextInt();

count++;

}

double average = sum / (double) NUM\_GRADES;

We combine the count we used in the while loop into the declaration of the for loop:

final int NUM\_GRADES = 10;

for( ; ; )

{

System.out.println(“Please enter the test grade”);

sum+=myScan.nextInt();

}

double average = sum/(double) NUM\_GRADES;

*Syntax:*

for ( initialization; loop condition; loop update )

{

// loop body

}

* Semicolons separate terms in the loop header
* no semicolon follows the loop header
* curly braces are required only if more than one statement is in the loop body

*Example 2: Print even numbers between 0 and 20* Flowchart:

Psuedocode:

for i = 0 to increased by

output i

Java:

*Example 3: Find Max of 10 grades*

Pseudocode: Java:

set max to a ridiculous (low) value

for i =

{

ask for a number

read the number

if the number is > max

set max to the number

}

print max

**Do-While Loop**

Checks the boolean expression AFTER the loop

*Example*: averaging 10 grades

Algorithm: Flowchart:

* Repeat 10 times:
  + Read in grade
  + Add grade to sum
* Find average (sum / 10)

Code:

**Loops with Strings** – What does this code do?

String s = JOptionPane.showInputDialog(null, “Please give a string:”);

int position = 0, counta = 0, countb = 0;

while ( position < s.length() )

{

if ( s.charAt (position) == ‘a’)

counta ++;

else if (s.charSt(position) == ‘b’)

countb ++;

position++;

}

**Practice**

How many times will each loop execute?

1. for (int i = 0; i < 10; i = i + 2)

{

//body of loop here

}

2. for (int k = 10; k >= 0; k = k-3)

{

  //body of loop here

}

3. int count = 0;

do

{

//body of loop here

  count++;

}while(count < 5);

4. int count=0;

while(count > 5)

{

  //body of loop here

count++;

}

***Nested Loops***

*Nested =*

**Example 1:**

for ( int i = 1; i < 20; i ++ )

{

for ( int j = 1; j < 5; j++ )

{

System.out.println ("Hi");

}

}

How many “Hi”s?

**Example 2:**

int count = 1, limit = 4;

finished = false;

while (!finished)

{

int fact = 1;

for (int i =1; i <= count; i++)

fact = fact \* i;

System.out.println (count + " " + fact);

if (count == limit)

finished = true;

else

count++

}

What does the code do?

What is the output?

*Side note: Nested loops will be very helpful when we process 2-dimensional arrays.*