**Some Additional Graphics Notes**

**Assignment**

<Variable> = <Expression>

d1 = 50

p1 = (150,150)

l1 = ((100,100),(200,200))

c1 = ((150,150),50)

The expression on the right hand side of the equals symbol is evaluated and the resulting value is placed into the variable on the left hand side of the equals symbol.

**Sequence**

The order of statements matters!

This program produces a RED circle, not a green one.

c1 = ((150,150),50)

draw (c1)

color(green)

This program produces a green circle.

c1 = ((150,150),50)

color(green)

draw (c1)

This program also produces a green circle.

color(green)

c1 = ((150,150),50)

draw (c1)

This program is invalid. It generates an error.

color(green)

draw (c1)

c1 = ((150,150),50)

This program is segment is invalid. It attempts to use “d1” before a value is assigned to “d1”.

p1 = (150,150)

c1 = (p1, d1)

d1 = 50

**Repetition (iteration)**

Repetition allows a program to do a set of actions over and over some number of times.

<set things up for the first case>

repeat <X> times

loop

<take care of the current case>

<prepare for the next case>

endloop

**Gaining an Intuitive Understanding of Code**

Circle moves about 2/3 of the way across the screen.

d1 = 10

d2 = 150

d3 = 10

Repeat 20 times

loop

c1= ((d1,d2),d3)

draw(c1)

erase(c1)

d1 = d1 + 10

endloop

Circle moves about 2/3 of the way across the screen. But circle moves FASTER

d1 = 10

d2 = 150

d3 = 10

Repeat **10** times

loop

c1= ((d1,d2),d3)

draw(c1)

erase(c1)

d1 = d1 + **20**

endloop

Circle moves about 2/3 of the way across the screen. But circle moves SLOWER

d1 = 10

d2 = 150

d3 = 10

Repeat **40** times

loop

c1= ((d1,d2),d3)

draw(c1)

erase(c1)

d1 = d1 + **5**

endloop

Circle moves all the way across the screen. (In other words, it moves farther)

d1 = 10

d2 = 150

d3 = 10

Repeat **30** times

loop

c1= ((d1,d2),d3)

draw(c1)

erase(c1)

d1 = d1 + 10

endloop

Circle moves about 2/3 of the way across the screen. But the circle is LARGER.

d1 = **20**

d2 = 150

d3 = **20**

Repeat 20 times

loop

c1= ((d1,d2),d3)

draw(c1)

erase(c1)

d1 = d1 + 10

endloop

NOTE: The current version of Watson Graphics Language handles increment and decrement differently than the previous version (implemented in the full lab as opposed to the ebook).

Increment – Current version

Format: <distanceVariable> = <distanceVariable> + <distanceConstant>

(where the two distance variables must be the same)

Examples: d1 = d1 + 5 (adds 5 to d1)

d2 = d2 + 10 (adds 10 to d2)

Increment – Previous version (still used in the Waston labs)

Format: increment(<distanceVariable>, <distanceConstant>)

Examples: increment(d1,5) (adds 5 to d1)

increment(d2, 10) (adds 10 to d2)

decrement – Current version

Format: <distanceVariable> = <distanceVariable> - <distanceConstant>

(where the two distance variables must be the same)

Examples: d1 = d1 - 5 (subtracts 5 from d1)

d2 = d2 - 10 (subtracts 10 from d2)

decrement – Previous version (still used in the Waston labs)

Format: decrement(<distanceVariable>, <distanceConstant>)

Examples: decrement(d1,5) (subtracts 5 from d1)

decrement(d2, 10) (subtracts 10 from d2)