**Review topics:**

Flow of control

Boolean expressions

Logical operators

Flow of Control—order of execution of statements

Order of statements can be

Linear

Selective

Repetitive

Boolean Expressions

Decisions that evaluate true or false can use relational operators

Logical expressions

! – logical not

&& - logical and

|| - logical or

**If statements and variants or them**

If only one statement do not need brackets, if more than 1 then enclose in brackets

if (condition)

{ Statement1;

}

Else

{ Statement2;

}

Conditions can be as simple or complicated as you make it.

**Nested if Statements**

Statement within another statement

If (condition)

{ statement;

}

Else

{

If (condition

{ Statement;

}

Else

{ Statement;

}

}

Can combine logical operators to eliminate nesting

If (x>=1 && x<=10)

Statement

Else

Statement

**While(loops)**

Executes a statement while it is true stops when the statement becomes false

Syntax:

While (condition)

Statement;

Need a loop control variable, while decide if it doesn’t run or runs infinitely

Int count = 1; //initial statement

While (count <=5) //condition

{

System.out.println(count);

Count ++; //update statement

}

If count = 10

The statement never executes and the loop is entirely skipped

If the update statement is changed to count--;

The loop will run forever because the numbers will always be less than 5

Sentinel value is a special value that marks the end of a list of values

**Switch Statement**

Switch and case are reserved words

Evaluates an expression then attempts to match to the results of several possible cases

Switch may use

Break – causes flow of control to transfer to the end

Default – causes control to transfer to it if no other cases match

If no default is called the first statement after the switch is chosen