Java for Android:
Indefinite Loops

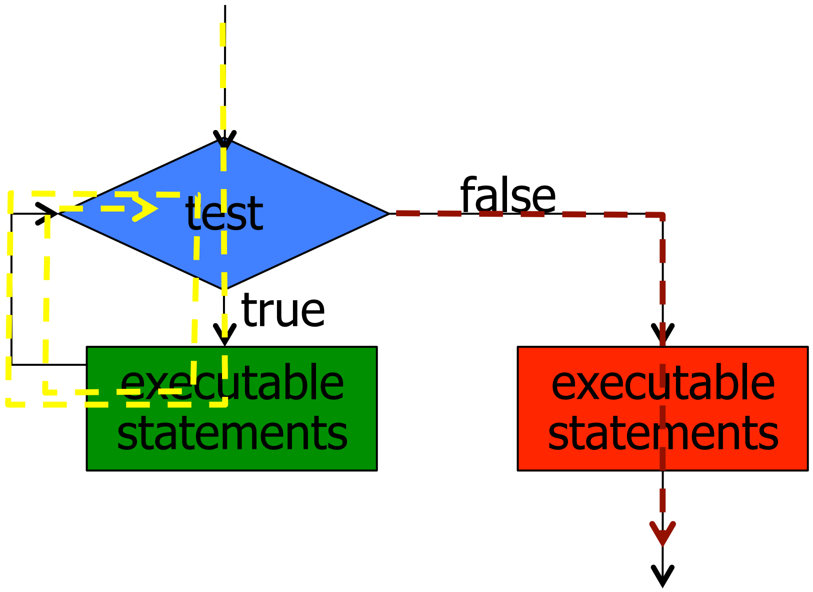
Julie L. Johnson

julie.L.johsnon@vanderbilt.edu

What if?

- What if I want to continually reduce some number n by 3 until the number is less than 19? How many times should my loop execute?
- What if I want to ask the user to keep guessing a number, and keep asking them until they guess correctly? How can I make this loop keep running?

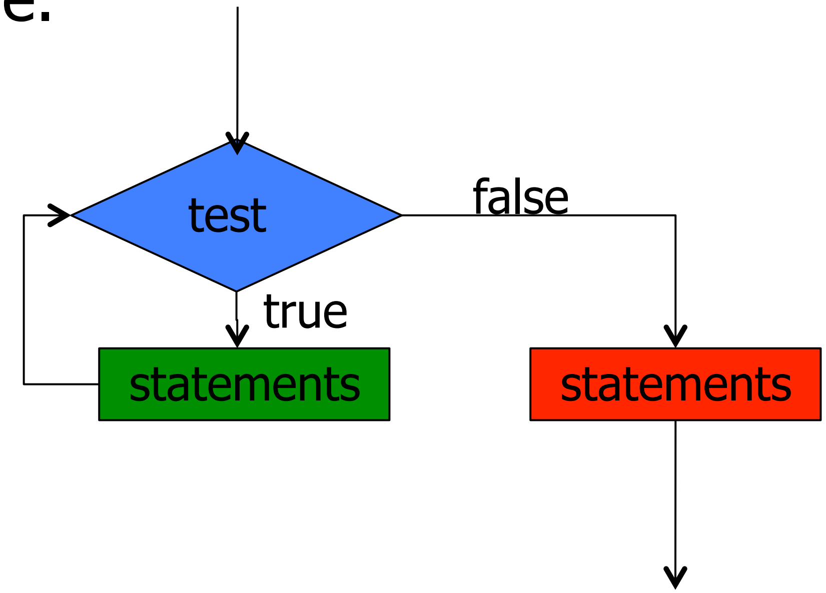
a loop that can run indefinitely



the while loop: syntax

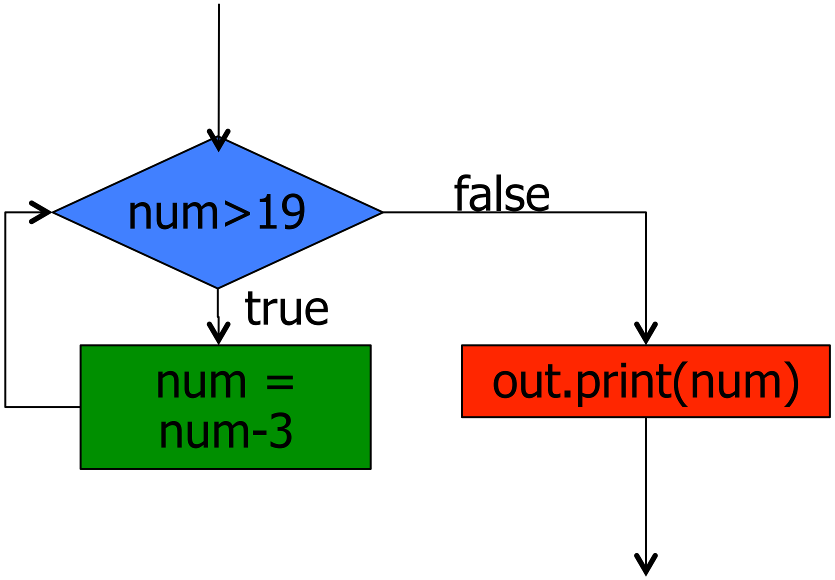
`while` loop: Repeatedly executes its body as long as a logical test is true.

```
while (test) {
     statement(s);
}
```



the while loop: an example

```
while (num > 19) {
     num = num - 3;
}
out.print(num);
```



the while loop and the for loop

The while loop can do all that the for loop can do

```
int total = 0;
for (int i = 0; i<5; i++){
        total = total + i;
}
```

```
int total = 0;
int i = 0;
while (i < 5){
        total = total + i;
        i++;
}
```

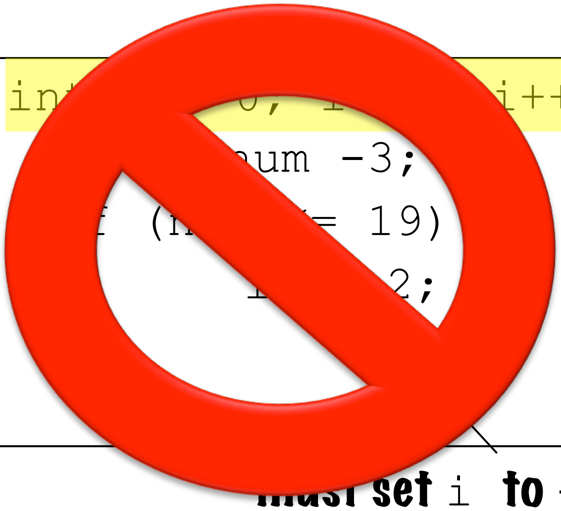
the while loop and the for loop

The for loop can do all that the while loop can do

but in a very clumsy way

```
for (int       0, i       i++){
                  num -3;
            f (n    = 19)
                 i    2;


}
```

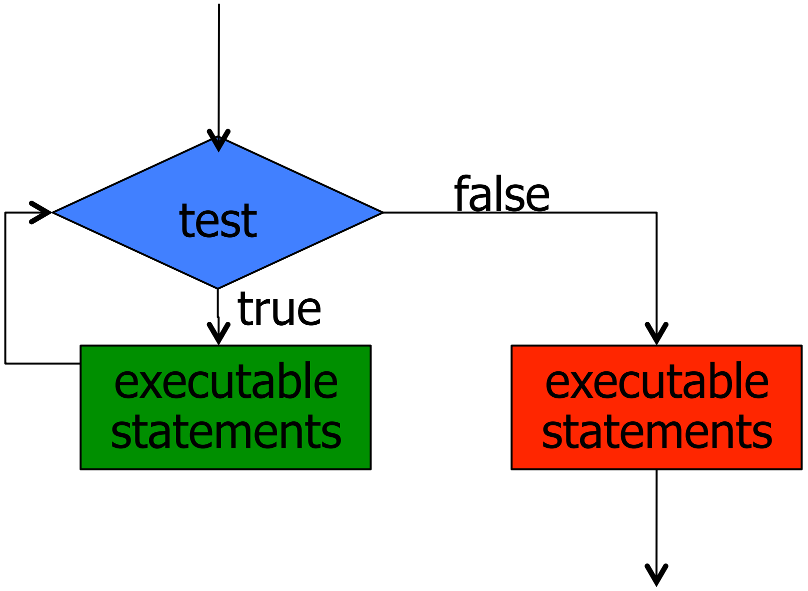


```
while (num > 19){
      num = num - 3;
}
```

must set i to -2, when i++ is executed before it is tested again, it will be equal to -1 and the loop will terminate

Adopt the practice of using a while loop when the number of iterations is not known before you begin

- *an indefinite loop*
- *best used when the number of iteration is unknown*

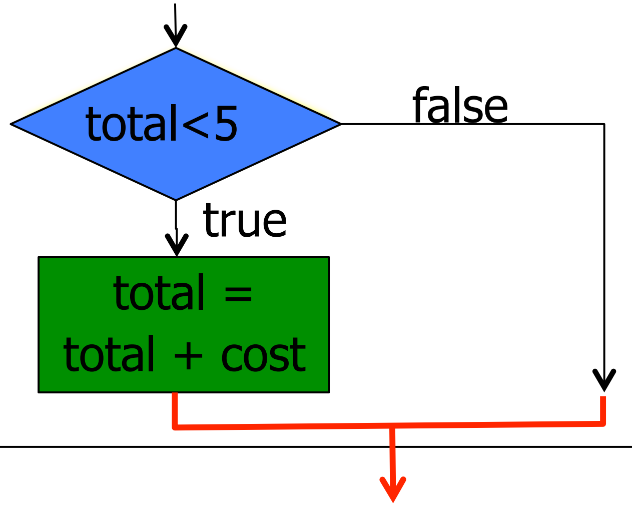


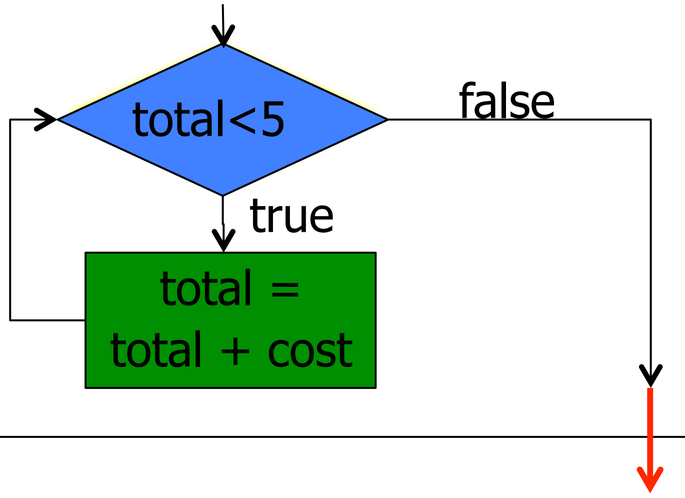
Notice that the syntax is similar to an if statement

but the result is very different!

```
int total = 0;
if (total < 5)
        total = total + cost;
}
```

```
int total = 0;
while (total < 5){
        total = total + cost;
}
```



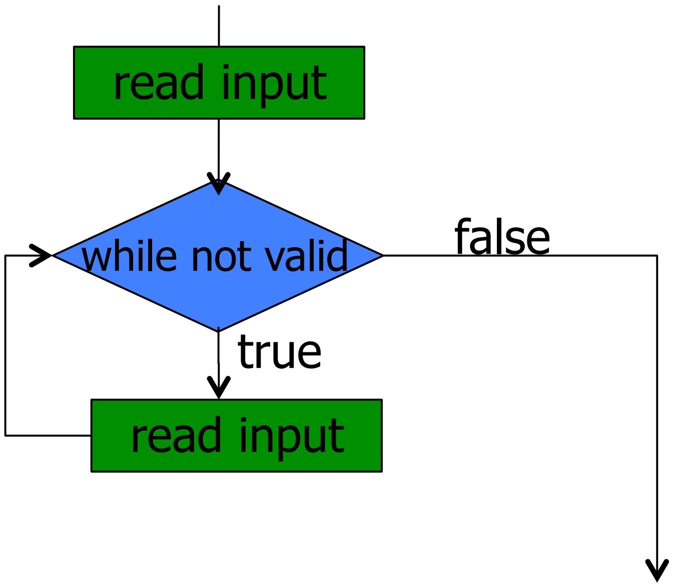


while loops are useful when testing user input

```
Out.print(“Enter a number between 0 and 5”)
int num = In.nextInt();
while ((num <0) || (num >5)){
    Out.print(“Your number is out of range.”);
    Out.print(“please try again: ”);
    num = In.nextInt();
}
```

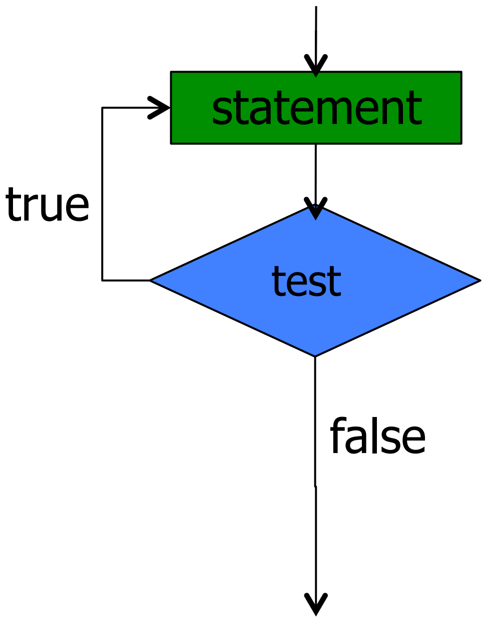
Checking user input

- *notice there was some redundancy in the code*
- *this can be avoided*



the do while loop

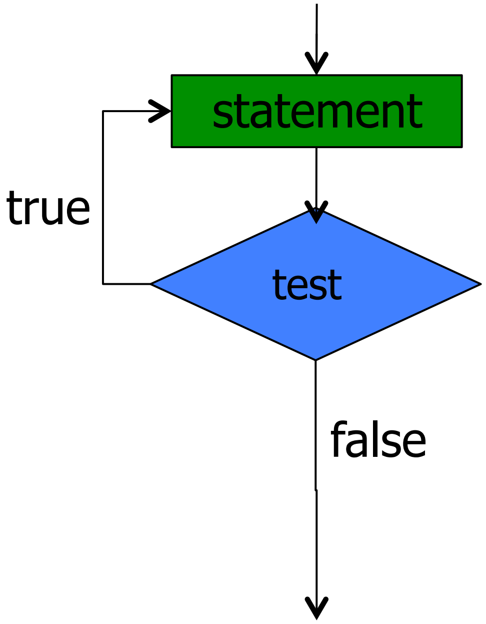
- *an indefinite loop*
- *best used when the number of iteration is unknown*
- *use when you will execute the loop at least once*



the do while loop: syntax

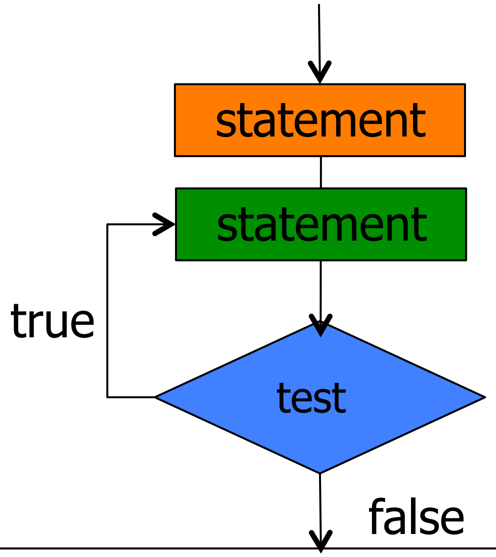
`do while` **loop**: executes its body and continues doing so as long as a logical test is true.

```
do {
     statement(s);
}while (test);
```

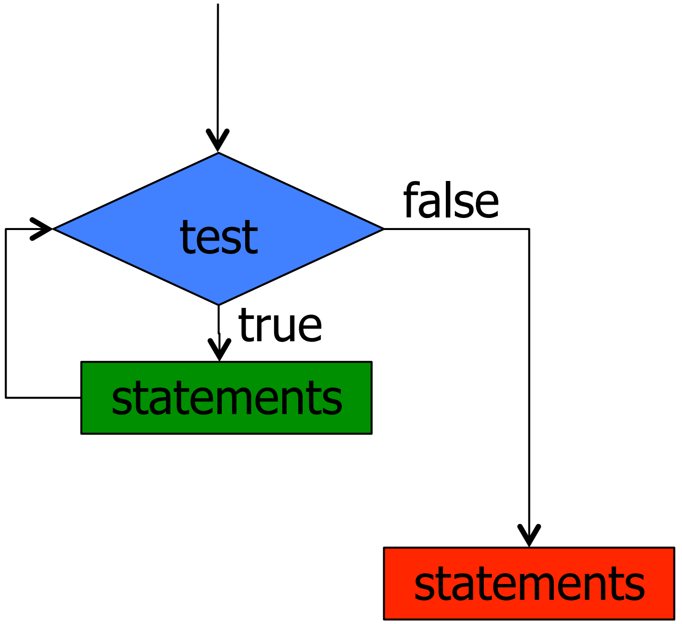


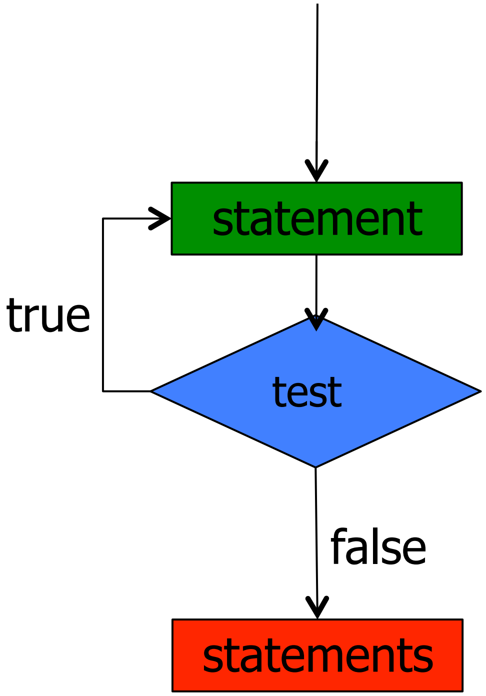
The do while loop: an example

```
if (num > 3){
    i = 1;
    Out.print(“Positive multiples of 3 less than the input”);
    do{
      Out.print(3*i);
      Out.print(“ ”);
      i++;
    }while (3*i < num);
}
```



indefinite loop constructs


test
false
true
statements
statements


statement
true
test
false
statements