Conditional and Repetitive Execution

Cumulative Sums
&
`if`
&
`if else`
&
`while`
&
`do while`

Adding many numbers

- Consider this code to read and add three values:

```
Scanner console = new Scanner(System.in);
System.out.print("Input a temperature: ");
double temperature1 = console.nextDouble();

System.out.print("Input a temperature: ");
double temperature2 = console.nextDouble();

System.out.print("Input a temperature: ");
double temperature3 = console.nextDouble();

System.out.print("Input a temperature: ");
double temperature4 = console.nextDouble();

System.out.print("Input a temperature: ");
double temperature5 = console.nextDouble();

double temperatures = temperature1 + temperature2 +
          temperature3 + temperature4 + temperature5;
System.out.println("The sum:" + temperatures);
```

A Cumulating Sum

- The variables `temperature1`, `temperature2`. . . `temperature5` are unnecessary:

```
Scanner input = new Scanner(System.in);
double sum = 0.0;

System.out.print("Input a temperature: ");
sum = sum + input.nextDouble();
System.out.print("Input a temperature: ");
sum = sum + input.nextDouble();

System.out.print("Input a temperature: ");
sum += input.nextDouble();

System.out.print("Input a temperature: ");
sum += input.nextDouble();

System.out.print("Input a temperature: ");
sum += input.nextDouble();

System.out.println("The sum:" + sum);
```

A Loop Reduces Redundancy

- An incorrect solution:

```
Scanner input = new Scanner(System.in);
for (int i = 1; i <= 5; i++) {
    int sum = 0;
    System.out.print(" Input a temperature:");
    sum += console.nextDouble();
}

System.out.println("The sum is " + sum);
```

  - The scope of `sum` is inside the `for` loop, so the last line of code fails to compile.

A Loop Reduces Redundancy

- A correct solution:

```
int sum = 0;
Scanner input = new Scanner(System.in);
for (int i = 1; i <= 5; i++) {
 System.out.print(" Input a temperature:");
    sum += console.nextInt();
}

System.out.println("The sum is " + sum);
```

  - Now the scope of `sum` is outside the `for` loop.

Variation: cumulative product

- The same idea can be used with other operators, such as multiplication which produces a cumulative product:

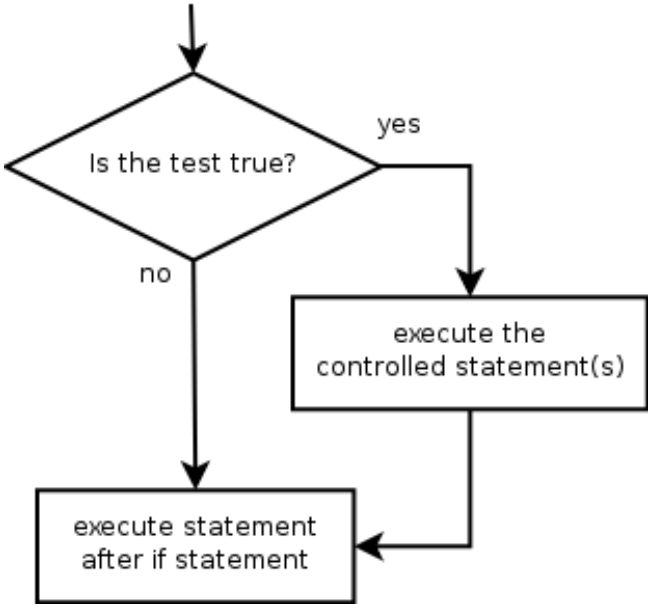
```
Scanner console = new Scanner(System.in);
System.out.print("Raise 2 to what power? ");
int exponent = console.nextInt();

int product = 1;
for (int i = 1; i <= exponent; i++) {
    product = product * 2;
}
System.out.println("2 to the " + exponent + " = " + product);
```

`if` and `if/else` statements

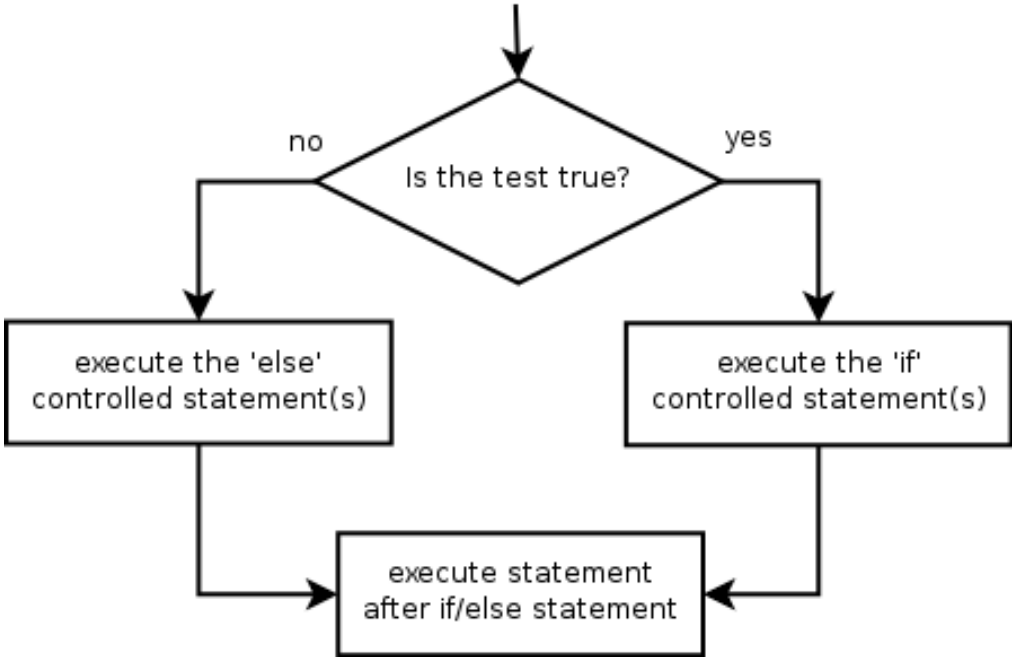
Textbook section: 4.2

Do it sometimes, but not always

`if` statement



`if / else` statement



Be careful: Don’t make a ‘dangling else’!!

The if statement

- **`if` statement**: A Java statement that executes a block of statements only if a certain condition is true.
  - General syntax:

```
if (<condition>) {
    <statement> ;
    <statement> ;
    ...
    <statement> ;
}
```

- Example:

```
double gpa = console.nextDouble();
if (gpa >= 2.0) {
    System.out.println("Your application is accepted.");
}
```

The if/else statement

- **`if/else` statement**: A Java statement that executes one block of statements if a certain condition is true, and a second block of statements if it is false.
  - General syntax:

```
if (<condition>) {
    <statement(s)> ;
} else {
    <statement(s)> ;
}
```

- Example:

```
double gpa = console.nextDouble();
if (gpa >= 2.0) {
    System.out.println("Welcome to Mars University!");
} else {
    System.out.println("Your application is denied.");
}
```

Example: Checking for Correct Input

```
Scanner input = new Scanner(System.in);
System.out.print("Input age ==>" );
int age = input.nextInt();

if (age < 5) {
        System.out.println("Toddler");
}
if (age < 15) {
        System.out.println("Child");
}
if (age < 20) {
        System.out.println("Teenager");
}
if (age < 30) {
        System.out.println("Young Adult");
}
if (age < 100) {
        System.out.println("Adult");
}
```

Doesn't Work Correctly!

Why ??

Example: Fixed

```
Scanner input = new Scanner(System.in);
System.out.print("Input age ==>" );
int age = input.nextInt();

if (age < 5) {
   System.out.println("Toddler");
}
else {
   if (age < 15) {
      System.out.println("Child");
   }
   else {
      if (age < 20) {
         System.out.println("Teenager");
      }
      else {
         if (age < 30) {
            System.out.println("Young Adult");
         }
         else {
            if (age < 100) {
               System.out.println("Adult");
            }
         }
      }
   }
}
```

That's Better!

Notice Indenting . .

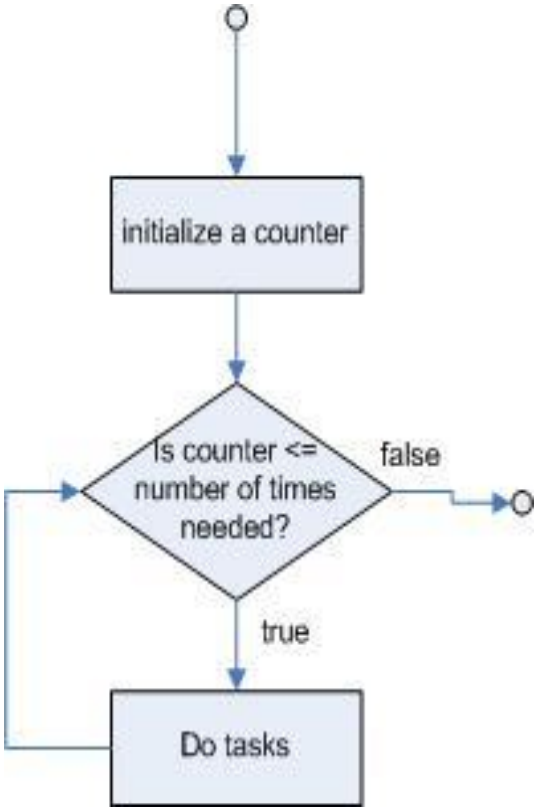
There is another acceptable way. . . .

Example: Alternate indent {}

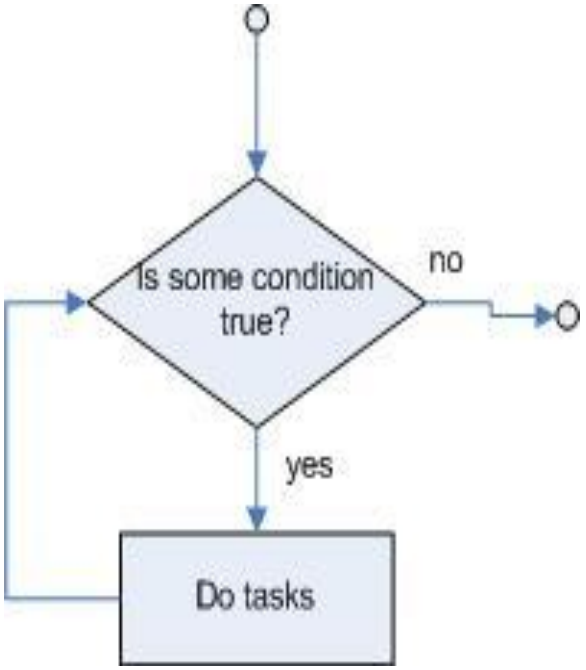
```
Scanner input = new Scanner(System.in);
System.out.print("Input age ==>" );
int age = input.nextInt();

if (age < 5) {
   System.out.println("Toddler");
}
else if (age < 15) {
   System.out.println("Child");
}
else if (age < 20) {
   System.out.println("Teenager");
}
else if (age < 30) {
   System.out.println("Young Adult");
}
else if (age < 100) {
   System.out.println("Adult");
}
else System.out.println(“Oldie Goldie”);
```

More Repetition

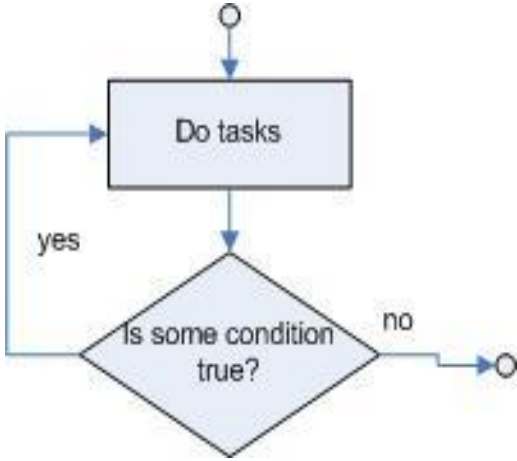
`for` loop



`while` loop

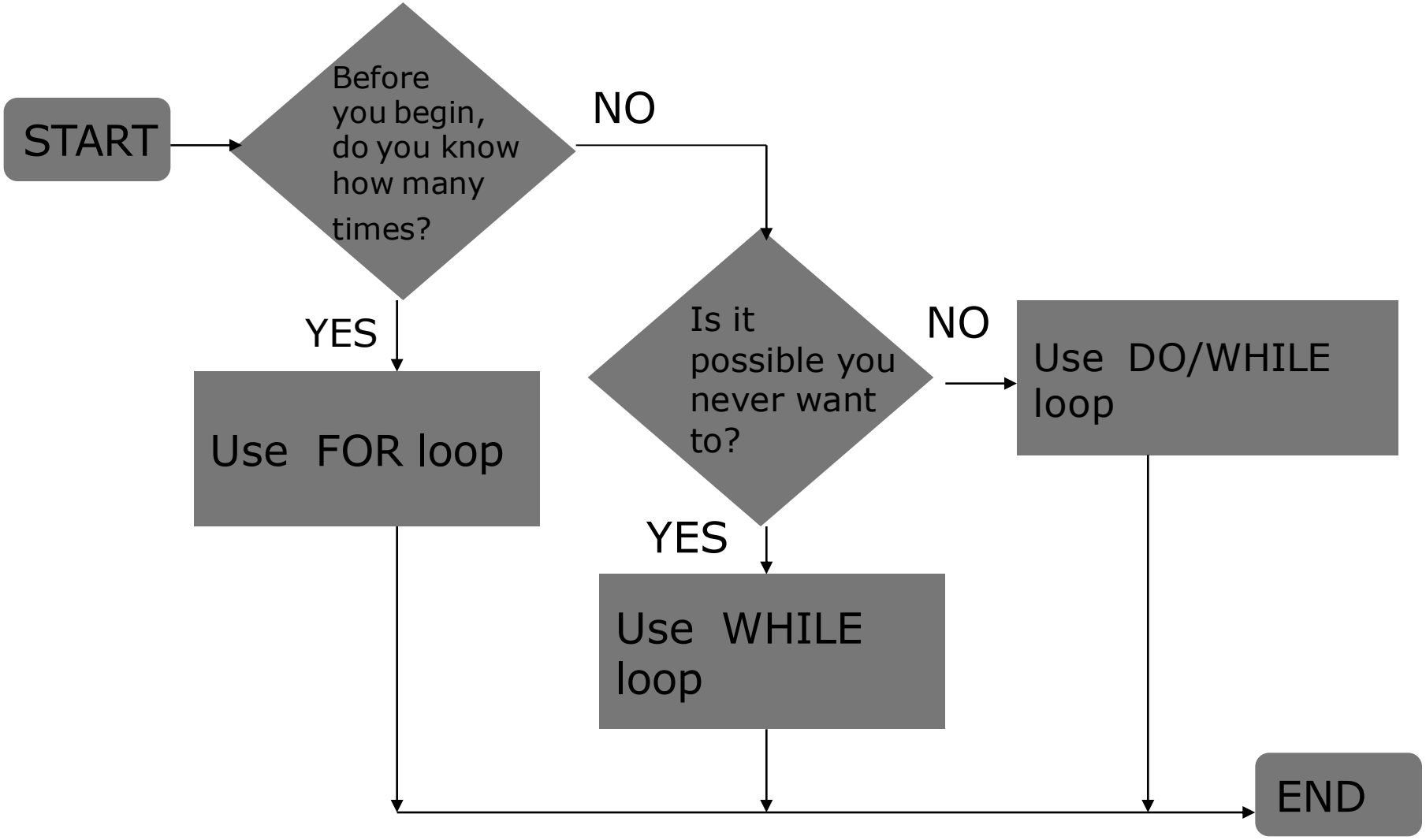


`do while` loop



When is it best to use which loop??

Doing it a LOT of times!!



A Problem:

Calculate the sum and product of a bunch of numbers. The user will type them in and indicate when she is done by typing a 0.

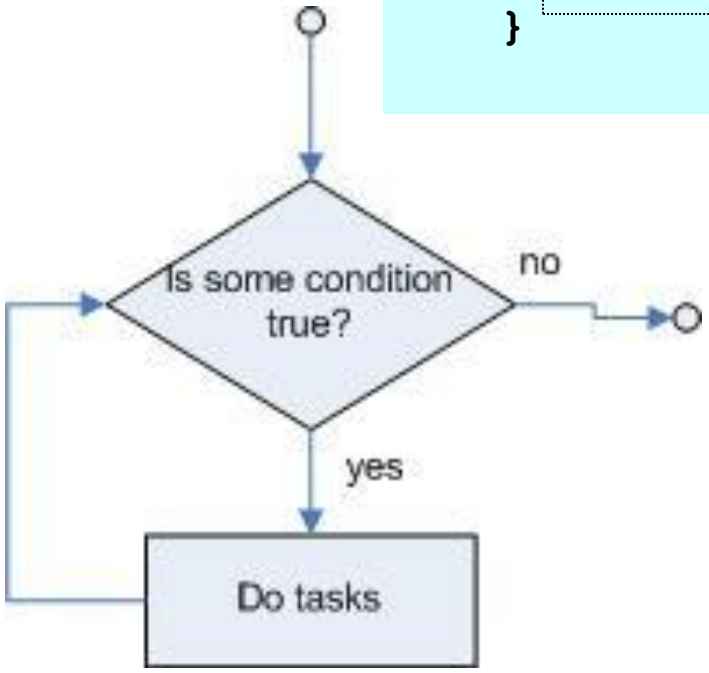
- That's called a ***sentinel***

- **The FOR loop is not suitable. . .**
- **Need to know the number BEFORE you start!!!**

Do you have a sentinel to indicate how many times to repeat?

Use a while loop:

```
int sum = 0;
int product = 1;
int inputNumber = stdin.nextInt();
while (inputNumber != 0) {
    sum = sum + inputNumber;
    product = product * inputNumber;
    inputNumber = stdin.nextInt();
}
```



- A cute Shortcut:

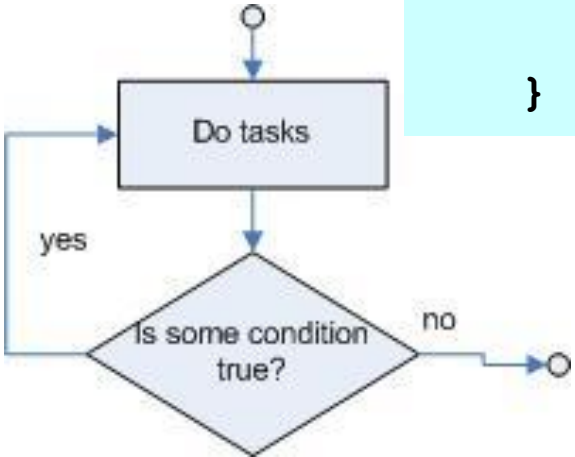
```
sum += inputNumber;
```

- `product *= inputNumber;`

When (exactly) should you check that condition?

To check at the end: use a do/while loop:

```
int sum = 0;
int product = 1;
int inputNumber;
do {
    inputNumber = stdin.nextInt();
    sum += inputNumber;
    if (inputNumber !=0) product *= inputNumber;
} while (inputNumber != 0);
```



Example: If or Loop??

```
Scanner input = new Scanner(System.in);
System.out.print("Input age ==>" );
int age = input.nextInt();

// What if a negative number is input?

if (age < 0) {
  System.out.println("Error: Positive number required");
  System.out.print("Input age ==>" );
  age = input.nextInt();
}


// Replace the above boxes with:

int age;
do {
  System.out.print("Input age ==>" );
  age = input.nextInt();
} while (age < 0);
```

Only works for 1 error only!