

## Making Choices (Chap 3)

- **If** you wish to defrost, press the defrost button; otherwise press the full power button.
- Let the dough rise in a warm place **until** it has doubled in size.

---

---

---

---

---

---

---

---

## Expressions and Statements

- Expression statements are formed by adding a semicolon to the end of certain types of expressions.
  - An assignment expression is an expression involving an assignment.  
`area = width * height;`
  - A method call expression has no assignment.  
`System.out.println(...);`

---

---

---

---

---

---

---

---

## Blocks

Several statements can be grouped into a block using { }.

```
{  
  x = 1;  
  {  
    y = 2;  
    System.out.println(y);  
  }  
  System.out.println(x);  
}
```

---

---

---

---

---

---

---

---

### A block with a declaration

```
{  
  int i = 5 + j;  
  // i is declared in this  
  // block j is from elsewhere  
  ...  
} // end of block, i disappears  
//can't access i here
```

---

---

---

---

---

---

---

---

### Boolean Expressions

- Any expression that evaluates to true or false.
- Relational operators, <, <=, >, >=.
- Equality operators, ==, !=.
- For example:  

```
int i = 3, j = 4;  
boolean flag;  
flag = 5 < 6;  
flag = (i == j);  
flag = (j + 2) <= 6;
```

---

---

---

---

---

---

---

---

### Logical Operators

- Operators that take boolean values as operands.
- x && y - true if x AND y are both true
- x || y - true if either x OR y are true, or both
- !x - true if x is false - read NOT x

---

---

---

---

---

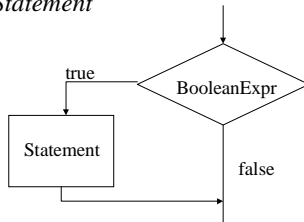
---

---

---

## The if statement

```
if ( BooleanExpression )  
    Statement
```



---

---

---

---

---

---

---

---

## Semicolons and the if statement

```
if (temperature < 32)  
    System.out.println("Warning: Below Freezing!");  
    System.out.println("It's " + temperature + "degrees");
```

```
if (temperature < 32) {  
    System.out.println("Warning Warning Warning!");  
    System.out.println("Warning: Below Freezing!");  
    System.out.println("Warning Warning Warning!");  
}
```

---

---

---

---

---

---

---

---

## Example: Sort three numbers

	Before		After
a	<input type="text" value="17"/>	a	<input type="text" value="6"/>
b	<input type="text" value="6"/>	b	<input type="text" value="11"/>
c	<input type="text" value="11"/>	c	<input type="text" value="17"/>

---

---

---

---

---

---

---

---

## Pseudocode

1. Place the numbers in boxes a, b, c
2. If the number in a is not larger than the number in b, go to 4.
3. Interchange the numbers in a and b.
4. If the number in b is not larger than the number in c, then go to 8.
5. Interchange the numbers in b and c.
6. If the number in a is not larger than the number in b then go to 8.
7. Interchange the numbers in a and b.
8. Halt

---

---

---

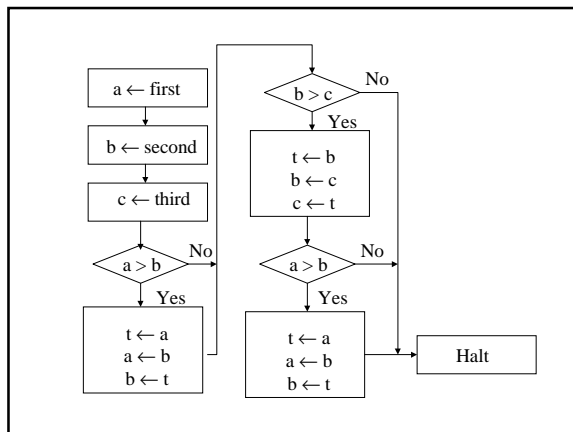
---

---

---

---

---



---

---

---

---

---

---

---

---

```
// SortInput.java - sort three numbers
import tio.*; // use the package tio

class SortInput {
    public static void main (String[] args) {
        int a, b, c, t;

        System.out.println("type three integers:");
        a = Console.in.readInt();
        b = Console.in.readInt();
        c = Console.in.readInt();
```

---

---

---

---

---

---

---

---

```
if (a > b) { // swap a and b
    t = a;
    a = b;
    b = t;
}
if (b > c) { // swap b and c
    t = b;
    b = c;
    c = t;
}
if (a > b) { // swap a and b
    t = a;
    a = b;
    b = t;
} // This doesn't match the pseudo-code. Why not?
System.out.print("The sorted order is : ");
System.out.println(a + ", " + b + ", " + c);
}
```

---

---

---

---

---

---

---

---

**The if-else statement**

```
if ( BooleanExpression )
    Statement1
else
    Statement2
```

```
graph TD
    Start(( )) --> BooleanExpr{BooleanExpr}
    BooleanExpr -- true --> Statement1[Statement1]
    BooleanExpr -- false --> Statement2[Statement2]
    Statement1 --> Join(( ))
    Statement2 --> Join
    Join --> End(( ))
```

---

---

---

---

---

---

---

---

**Finding the minimum**

```
if (x < y)
    min = x;
else
    min = y;
```

---

---

---

---

---

---

---

---

### A block in an if-else statement

```
if (temperature < 32) {  
    System.out.println("Warning");  
    System.out.println(32 - temperature +  
        "(F) below Freezing!");  
    System.out.println("Warning");  
}  
else {  
    System.out.println("It's " + temperature +  
        "degrees fahrenheit.");  
}
```

---

---

---

---

---

---

---

---

### Common Programming Error

What is printed by the following if x is 3 and y is 4?

```
if (x < y);  
    System.out.println(x + " is smaller");  
if (y < x);  
    System.out.println(y + " is smaller");
```

---

---

---

---

---

---

---

---

### Common Programming Error

What is printed by the following if temperature is 45?

```
if (temperature < 32)  
    System.out.println("It is now");  
    System.out.print(32 - temperature);  
    System.out.println(" below freezing.");  
System.out.println("It's " + temperature +  
    "degrees.");
```

---

---

---

---

---

---

---

---

### Nested if-else statements

```
if (ageOfPerson >= 18)
  if (ageOfPerson < 65)
    System.out.println("full fare adult");

if (ageOfPerson >= 18 && ageOfPerson < 65)
  System.out.println("full fare adult");
```

---

---

---

---

---

---

---

---

### Nested if-else statements

```
if (temperature < 32) {
  System.out.println("Warning");
  if (temperature < 0)
    System.out.println((-temperature) +
      "(F) below zero!");
  else
    System.out.println(32 - temperature +
      "(F) below Freezing!");
  System.out.println("Warning");
}
else {
  System.out.println("It is " + temperature);
}
```

---

---

---

---

---

---

---

---

### Common Programming Error

```
if (18 <= age < 65)
  ...

This is not legal syntax.
```

---

---

---

---

---

---

---

---

### if-else-if-else-if-else...

```
if (ageOfPerson < 18)
    System.out.println("child fare");
else {
    if (ageOfPerson < 65)
        System.out.println("adult fare");
    else
        System.out.println("senior fare");
}
```

---

---

---

---

---

---

---

### if-else-if-else-if-else...

```
if (ageOfPerson < 18)
    System.out.println("child fare");
else if (ageOfPerson < 65)
    System.out.println("adult fare");
else
    System.out.println("senior fare");
```

---

---

---

---

---

---

---

### if-else-if-else-if-else...

```
if (ageOfPerson < 18)
    System.out.println("child fare");
else if (ageOfPerson < 65)
    System.out.println("adult fare");
else
    System.out.println("senior fare");
```

---

---

---

---

---

---

---



## The dangling else problem

```
if (Expression1)
  if (Expression2)
    Statement1
else
  Statement2
```

This indentation is misleading.

---

---

---

---

---

---

---

---

## The dangling else problem

```
if (Expression1)
  if (Expression2)
    Statement1
  else
    Statement2
```

An else always matches the nearest unmatched if.

---

---

---

---

---

---

---

---

## The dangling else problem

```
if (Expression1){
  if (Expression2)
    Statement1
}
else
  Statement2
```

Use braces to force the grouping you want.

---

---

---

---

---

---

---

---

### The while statement

`while ( BooleanExpression )`  
`Statement`

Execution enters the loop

Continue with the rest of the program

---

---

---

---

---

---

---

---

### A simple loop

```

class Valentine {
    public static void main(String[] args) {
        int howMuch = 0;
        while (howMuch < 5) {
            System.out.println("I love you.");
            howMuch++;
        }
    }
}

```

---

---

---

---

---

---

---

---

### Average - pseudocode - no while

1. Get a number.
2. If the number is 0 go to step 6.
3. Add the number to the running total.
4. Increment the count of numbers read in.
5. Go back to step 1.
6. Divide the running total by the count of number in order to get the average.
7. Print the average.

---

---

---

---

---

---

---

---

## Average - pseudocode - using while

```
Get a number.  
While the number is not 0 do the following:  
    Add the number to the running total.  
    Increment the count of numbers read in.  
    Get a number.  
(when the loop exits)  
Divide the running total by the count of  
number in order to get the average.  
Print the average.
```

---

---

---

---

---

---

---

---

## General form of a while loop

```
InitializationStatement  
while ( BooleanExpression ) {  
    Statement1  
    Statement2  
    ...  
    PrepareForNextIteration  
}
```

---

---

---

---

---

---

---

---

## Average.java

- Dissect Average.java
- Identify parts of the loop - init, test, nextIter
- Sentinel controlled loop
- 0 / 0 is NaN

---

---

---

---

---

---

---

---

```
// Average.java - compute average of input values
import tio.*;

public class Average {
    public static void main(String[] args) {
        double number;
        int count = 0;
        double runningTotal = 0;
        // initialization before first loop iteration
        System.out.println("Type some numbers, "+
            "the last one being 0");
```

---

---

---

---

---

---

---

---

```
        number = Console.in.readDouble();
        while (number != 0) {
            runningTotal = runningTotal + number;
            count = count + 1;
            // prepare for next iteration
            number = Console.in.readDouble();
        }
        System.out.print("The average of the ");
        System.out.print(count);
        System.out.print(" numbers is ");
        System.out.println(runningTotal/count);
    }
}
```

---

---

---

---

---

---

---

---

## Common Programming Error

How many times does this loop execute?

```
int count = 13;
System.out.println("The multiples of " +
    "13 between 1 and 100 are:");
while (count != 100) {
    System.out.println(count);
    count = count + 13;
}
```

---

---

---

---

---

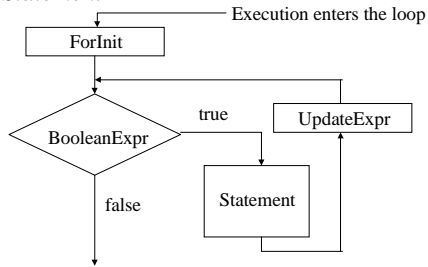
---

---

---

## The for statement

for ( *ForInit*; *BooleanExpr*; *UpdateExpr* )  
*Statement*



Continue with the rest of the program

---

---

---

---

---

---

---

---

```
// SquareRoots.java - print square roots of 1 - 10
public class SquareRoots {
    public static void main(String[] args) {
        int i;
        double square_root;

        for (i = 1; i <= 10; i++) {
            square_root = Math.sqrt(i);
            System.out.println("the square root of " + i +
                " is " + square_root);
        }
        System.out.println("That's All!");
    }
}
```

---

---

---

---

---

---

---

---

```
// SquareRoots.java - print square roots of 1 - 10
public class SquareRoots {
    public static void main(String[] args) {

        double square_root;

        for (int i = 1; i <= 10; i++) {
            square_root = Math.sqrt(i);
            System.out.println("the square root of " + i +
                " is " + square_root);
        }
        System.out.println("That's All!");
    }
}
```

---

---

---

---

---

---

---

---

```
// SquareRoots2.java - replace for with while
public class SquareRoots2 {
    public static void main(String[] args) {
        int i;
        double square_root;

        i = 1; // initialization-expr
        while (i <= 10) {
            square_root = Math.sqrt(i);
            System.out.println("the square root of " + i +
                " is " + square_root);

            i++; // iteration-expr
        }
        System.out.println("That's All!");
    }
}
```

---

---

---

---

---

---

---

---

## Switch

- Day of week with if-else
- Day of week with a switch
- Day of week grouping weekdays and weekends.

---

---

---

---

---

---

---

---

```
if (dayOfWeek == 1)
    System.out.println("Sunday");
else if (dayOfWeek == 2)
    System.out.println("Monday");
else if (dayOfWeek == 3)
    System.out.println("Tuesday");
else if (dayOfWeek == 4)
    System.out.println("Wednesday");
else if (dayOfWeek == 5)
    System.out.println("Thursday");
else if (dayOfWeek == 6)
    System.out.println("Friday");
else if (dayOfWeek == 7)
    System.out.println("Saturday");
else
    System.out.println("Not a day number " +
        dayOfWeek);
```

---

---

---

---

---

---

---

---

```

switch (dayOfWeek) {
    case 1:
        System.out.println("Sunday");
        break;
    case 2:
        System.out.println("Monday");
        break;
    // cases 3-6 omitted to make fit on one slide
    case 7:
        System.out.println("Saturday");
        break;
    default:
        System.out.println("Not a day number " +
            dayOfWeek);
        break;
}

```

---

---

---

---

---

---

---

---

```

switch (dayOfWeek) {
    case 1:
    case 7:
        System.out.println("Stay home today!");
        break;
    case 2:
    case 3:
    case 4:
    case 5:
    case 6:
        System.out.println("Go to work.");
        break;
    default:
        System.out.println("Not a day number " +
            dayOfWeek);
        break;
}

```

---

---

---

---

---

---

---

---

## A Final Example

Write a program to plot an increasing frequency sine wave, given an initial step size in degrees, the number of points to plot, and the rate of increase as a percentage.

```

class PlotSin {
    public static void main(String[] args) {
        <<Get initial step, num points, and rate>>
        while <<More lines to plot>>
            <<Plot the next point>>
    }
}

```

---

---

---

---

---

---

---

---

```
<<More lines to plot>>=  
  
(linesRemaining > 0)  
  
<<Plot the next point>>=  
  
{  
  <<Print some spaces>>  
  System.out.println('*');  
  <<Update the variables>>  
}
```

---

---

---

---

---

---

---

```
<<Print some spaces>>=  
  
// change -1 to 1 into 0 to 78  
int numSpaces =  
    Math.sin(Math.toRadians(angle)) * 39 + 39;  
while (numSpaces > 0) {  
    System.out.print(' ');  
}  
  
<<Update the variables>>=  
  
linesRemaining--;  
angle = angle + step;  
step = step * rate;
```

---

---

---

---

---

---

---

```
<<Get initial step, num points, and rate>>=  
  
double step, angle = 0;  
int pointsRemaining;  
  
System.out.println("Enter initial step size.");  
step = Console.in.readDouble();  
  
System.out.println("Enter number of points.");  
pointsRemaining = Console.in.readInt();  
  
System.out.println("Enter rate of increase.");  
rate = Console.in.readDouble();  
//Convert to 1 + fractional rate  
rate = 1 + rate / 100;
```

---

---

---

---

---

---

---



```
import tio.*;

class PlotSin {
    public static void main(String[] args) {
        //get the step size, number of lines, and rate of increase
        double step, rate, angle = 0;
        int pointsRemaining;

        System.out.println(
            "Enter the initial step size in degrees.");
        step = Console.in.readDouble();

        System.out.println("Enter the number of points to plot.");
        pointsRemaining = Console.in.readInt();

        System.out.println(
            "Enter the rate of increase as a percentage.");
        rate = Console.in.readInt();
        rate = 1.0 + (rate / 100);
    }
}
```

---

---

---

---

---

---

---

---

```
while (pointsRemaining > 0) {
    // print some spaces
    // change -1 to +1 range of sin() into 0 to 78.
    int numSpaces = (int)Math.round(
        Math.sin(Math.toRadians(angle)) * 39 + 39);
    while (numSpaces > 0) {
        System.out.print(' ');
        numSpaces--;
    }

    System.out.println('*'); // draw the point

    // update the variables for the next iteration
    pointsRemaining--;
    angle = angle + step;
    step = step * rate; // incr step => incr frequency
}
}
```

---

---

---

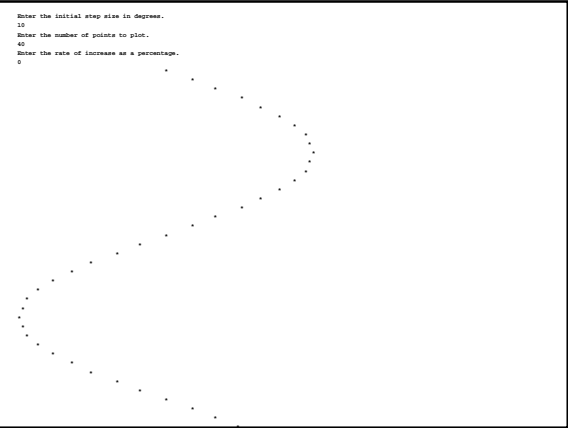
---

---

---

---

---



---

---

---

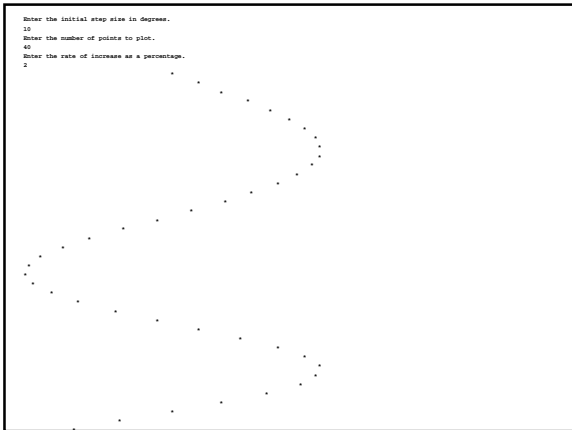
---

---

---

---

---



---

---

---

---

---

---

---