

Conditionals and Loops

Nesting



Nested If Statements

Ex. Pay a certain tax rate depending on income level.

| Income | Rate |
|-------------------|------|
| 0 - 47,450 | 22% |
| 47,450 - 114,650 | 25% |
| 114,650 - 174,700 | 28% |
| 174,700 - 311,950 | 33% |
| 311,950 - | 35% |

5 mutually exclusive alternatives

```
double rate;  
if      (income < 47450) rate = 0.22;  
else if (income < 114650) rate = 0.25;  
else if (income < 174700) rate = 0.28;  
else if (income < 311950) rate = 0.33;  
else                                     rate = 0.35;
```

graduated income tax calculation

Nested Loops

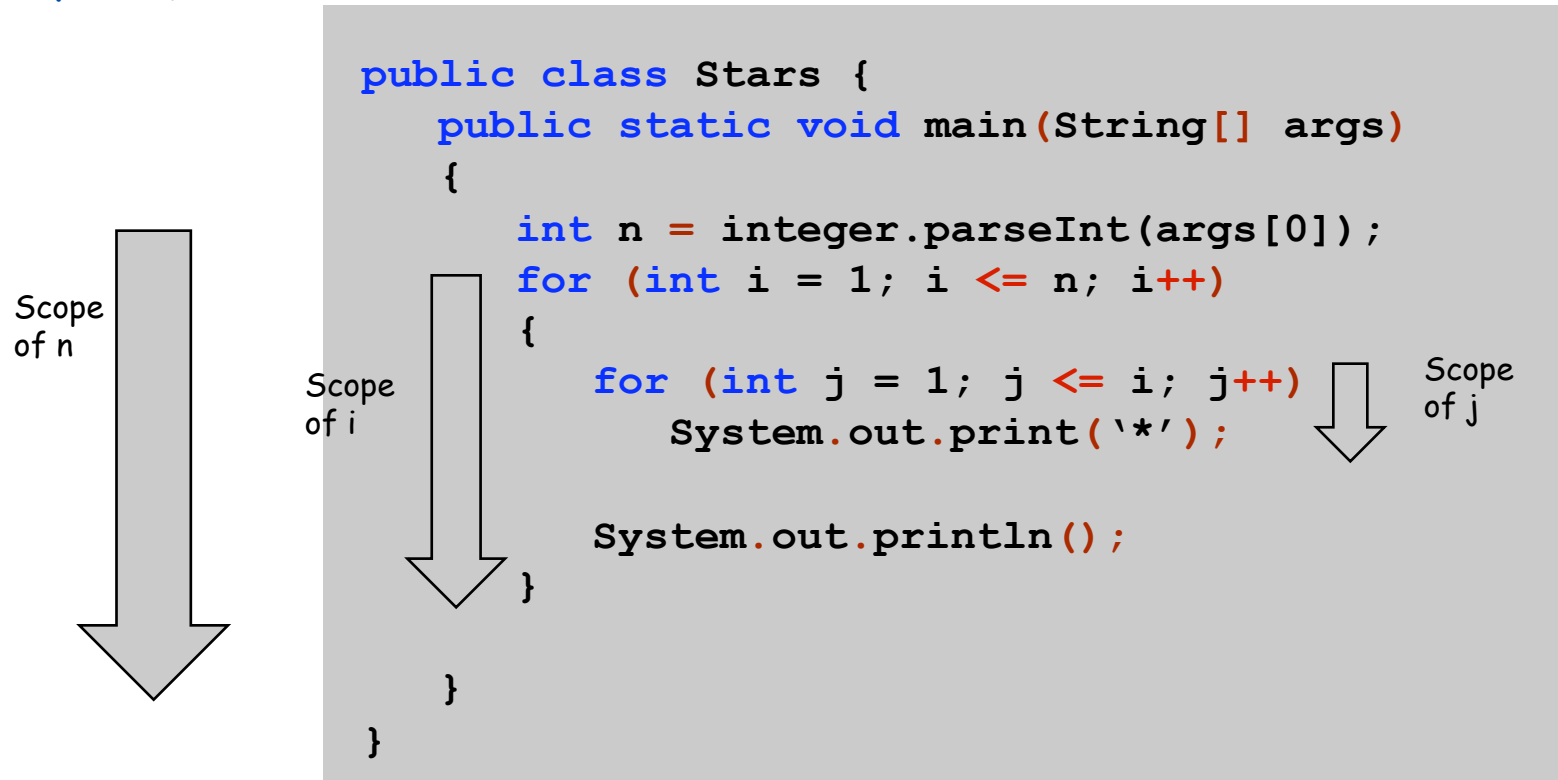
Ex. Print the following pattern on the screen

```
*  
**  
***  
****  
*****
```

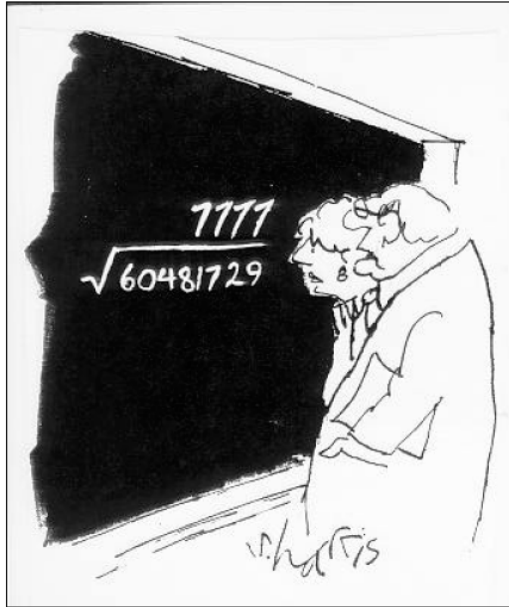
```
public class Stars {  
    public static void main(String[] args)  
    {  
        int n = Integer.parseInt(args[0]);  
        for (int i = 1; i <= n; i++)  
        {  
            for (int j = 1; j <= i; j++)  
                System.out.print('*');  
  
            System.out.println();  
        }  
    }  
}
```

Variable Scopes

The scope of variable is the range of a program where the variable is defined.



A Wonderful Square Root



"A wonderful square root. Let's hope it can be used for the good of mankind."

Copyright 2004, Sidney Harris, <http://www.sciencecartoonsplus.com>

```
% java Sqrt 60481729  
7777.0
```

While Loops: Square Root

Q. How might we implement `Math.sqrt()` ?

A. To compute the square root of c :

- Initialize $t_0 = c$.
- **Repeat until** $t_i = c / t_i$, up to desired precision:
set t_{i+1} to be the average of t_i and c / t_i .

$$\begin{aligned}t_0 &= 2.0 \\t_1 &= \frac{1}{2} \left(t_0 + \frac{2}{t_0} \right) = 1.5 \\t_2 &= \frac{1}{2} \left(t_1 + \frac{2}{t_1} \right) = 1.4166666666666665 \\t_3 &= \frac{1}{2} \left(t_2 + \frac{2}{t_2} \right) = 1.4142156862745097 \\t_4 &= \frac{1}{2} \left(t_3 + \frac{2}{t_3} \right) = 1.4142135623746899 \\t_5 &= \frac{1}{2} \left(t_4 + \frac{2}{t_4} \right) = 1.414213562373095\end{aligned}$$

computing the square root of 2

While Loops: Square Root

Q. How might we implement `Math.sqrt()` ?

A. To compute the square root of c :

- Initialize $t_0 = c$.
- **Repeat until** $t_i = c / t_i$, up to desired precision:
set t_{i+1} to be the average of t_i and c / t_i .

```
public class Sqrt {
    public static void main(String[] args) {
        double EPS = 1E-15;
        double c = Double.parseDouble(args[0]);
        double t = c;
        while (Math.abs(t - c/t) > t*EPS) {
            t = (c/t + t) / 2.0;
        }
        System.out.println(t);
    }
}
```

error tolerance

```
% java Sqrt 2.0
1.414213562373095
```

15 decimal digits of accuracy in 5 iterations

Applications of Conditionals and Loops

- Code Construction and pseudo-code.
- Programming examples:
 - (1) Compute the binary representation of an integer number.
 - (2) Factor a number into the product of a sequence of (prime) numbers.