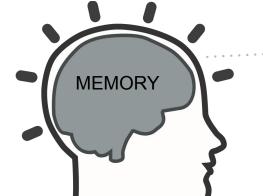
Week 1

Basic Patterns

What is programming?

Programming is the process of converting a specification into a program.



Patterns

process

Specification

Program

A description of a program's goals

> Code that achieves the goals

What is the process?

Not a single process!

- Words to code table
- Key/framework approach
- Debugging/Testing
- Break it down, build it up
- Class diagrams
- Location tables
- + a bit of creativity and intuition^[1]

[1] A classic text on creativity: "Lateral Thinking" by Edward De Bono (1970)

What is a pattern?

A common solution to a common problem.

Example program:

```
int totalRainfall = 0;
for (int i = 0; i < rainfall.length; i++)
{
    totalRainfall += rainfall[i];
}</pre>
```

The pattern: "sum"

```
sum = 0;
for each item
{
    sum += item;
}
```

The "sum" pattern

Goal: Find the sum of a collection of items.

```
<type> sum = 0;
<for each item>
{
    sum += <item>;
}
```

- Words enclosed between
 <angled brackets> are holes
 that need to be filled.
- <type> is typically double or int
- <for each item> is any loop over a collection of items
- <item> refers to the next item in the loop.
- x += y adds the amount y onto x.

The "output" pattern

Goal: Show a value to the user.

Pattern:

```
System.out.println("<label>" + <value>);
```

• e.g. show an age:

```
System.out.println("age is " + age);
```

• e.g. show a name:

```
System.out.println("name is " + name);
```

The "read" pattern

Goal: Read a value from the user.

Pattern:

```
System.out.print(""rompt>");
<type> <variable> = <read operation>;
Or,
System.out.print("prompt>");
<variable> = <read operation>; (if <variable> has already been declared)
• e.g. read an age:
System.out.print("Age: ");
int age = In.nextInt();
```

Read operations

We provide you with class In which has 4 static methods:

public static int nextInt()	Reads an integer
public static double nextDouble()	Reads a double
public static char nextChar()	Reads a single character
public static String nextLine()	Reads a String

Copy this class into your project.

Then simply call the method on the class: In.nextInt()

The "read loop" pattern

Goal: Read values until the user enters an "end of input" value.

```
<read pattern>
while (<value> != <end value>)
{
      <use the value>
      <read pattern>
}
```

Observations:

- <read pattern> appears twice
- always test for the "end of input" value immediately after a <read pattern>.

The "array loop" pattern

Goal: Loop over items in an array.

```
for (int i = 0; i < <array>.length; i++)
{
      <use the item array[i]>
}
```

DEM₀

DEM₀

The "count" pattern

Goal (without guard): Count the number of items in a collection. **Goal** (with guard): Count the number of items that satisfy a condition.

Without guard:

```
int count = 0;
<for each item>
    count++;
```

With guard:

```
int count = 0;
<for each item>
   if (<guard>)
        count++;
```

X++ VS ++X

- Both x++ and ++x add 1 to x.
- x++ returns the value of x and THEN adds 1 to x.
- ++x adds 1 to x and THEN returns the value of x.

e.g.

```
int x = 7;

System.out.println(x++); // shows

7
x = 8
int x = 7;

System.out.println(++x); // shows

8
```

DEM₀

The "max" pattern

Goal: Find the maximum value in a collection of items

```
<type> max = <smallest number>;
<for each item>
{
    if (<item> > max)
        max = <item>;
}

Key idea:

If this item is bigger than the max so far,
then make it the new max.
```

Process

Specification: Read in card numbers until the user enters -1. Show the highest card.

```
Enter card number: 8
Enter card number: 3
Enter card number: 12
Enter card number: 7
Enter card number: -1
The highest card is 12
```

Specification: Read in card numbers until the user enters -1. Show the highest card.

Break the specification down into pieces:

Read in the card numbers /

until the user enters -1 /

Show /

the highest card

Arrange the words into a table:

Words	Code/pattern
Read in card numbers	
until the user enters -1	
Show	
the highest card	

Translate the words to code:

Words	Code/pattern
Read in card numbers	read
until the user enters -1	read loop
Show	output
the highest card	max

DEM₀