RADFORD UNIVERSITY

ITEC 120

Lecture 12
Arrays

Review

- Questions?
- Software Design and Testing
  - What are they?
  - What do we use them for?

Objectives

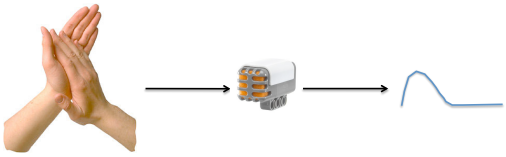
- Rationale for arrays
- Basic syntax
- Examples

Variables

If one is good
More is better

Problem

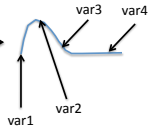
- Data sensor that produces 100 whole numbers a second
- Need to perform calculations on them



Arrays

Problem

```
int var1,var2,var3,var4;
```



-Painful if you have large amounts of data to work on
-Need a way to refer to lots of variables with one variable

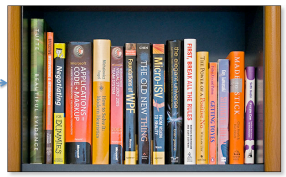
Arrays

Idea

Two tiered reference mechanism (SAME TYPE OF INFO)

On the top bookshelf

Book 0 => Beautiful Evidence
Book 1 => Effective C++
Book 2=> Negotiating
Book 3=> Applications



Arrays

Visual example

- ArrayTeacher
  - Available on website

Arrays

Java

```
int[] values = null;
```

Empty array

FIXED SIZE

- Array
  - Type
  - Size

Type Name

```
int[] values = new int[10];
```

It is an array

Make space for 10 integers

- Storing / Accessing

```
int num=0;
values[0] = 30;
System.out.println(values[0]);
values[num] = 30;
values[values.length-1]=30;
```

Arrays

Accessing

```
int[] example = new int[5];
```

- First element `example[?];`
- Last element `example[?];`
- Middle element `example[?];`

Arrays

*Candy example

Storing

```
int[] example = new int[5];
```

```
example[0] = 2;
example[1] = 3;
example[2] = 4;
example[3] = 5;
example[4] = 6;
```

Arrays

Strings

```
String a = "Example";
char[] b = new char[a.length()];
b[0] = a.charAt(0);
b[1] = a.charAt(1);
b[2] = a.charAt(2);
b[3] = a.charAt(3);
b[4] = a.charAt(4);
b[5] = a.charAt(5);
b[6] = a.charAt(6);
```

Arrays

Functions

```
int getFirst(int[] data)
{
    if (data.length > 0)
    {
        return data[0];
    }
    return 0;
}
```

Parameter

```
public int getValue(int[] data, int index)
{
    if (data.length > 0 && index <data.length)
    {
        return data[index];
    }
    return 0;
}
```

Arrays

Loops

```
int[] values = new int[10];
```

Possible indexes

```
values[0] values[1] values[2] values[3] values[4]
values[5] values[6] values[7] values[8] values[9]
```

```
for (int i=0; i<10; i++)
{
   System.out.println(i);
}
```

```
for (int i=0; i<values.length; i++)
{
    values[i] = i;
}
```

Arrays

Example

- Read in X numbers
- Find min

```
int x = scan.nextInt();
int[] array = new int[x];
for (int i=0; i<x; i++)
{
    array[i] = scan.nextInt();
}
int min=0;
for (int i=1; i<array.length; i++)
{
    if (array[i] < array[min])
        min = i;
}
```

Arrays

Find average

- Read in X numbers, print out average
- jEdit example

Arrays

Counting #s

- Generate numbers between 0-50
- Count how many of each number we have
- What problems could this be applied to?

Random array

```
int x = random.nextInt(20);
int array = new int[x];
for (int i=0; i<x; i++)
{
   array[i] = random.nextInt(100);
}
int zeroToSixty=0;
int sixtyOrGreater=0;
for (int i=0; i<array.length; i++)
{
   if (array[i] > 0 && array[i] <= 60)
   {
      zeroToSixty++;

   }
   else if (array[i] > 60)
      sixtyOrGreater++;
}
```


Array Visualization

```
 _____________
|0 to 60    | *****    |
|61 to 100  | ********|
```

```
System.out.println("_____________");
System.out.print(   "| 0 to 60 |");
for (int i=0; i<  zeroToSixty; i++)
{
   System.out.print("*");
}
System.out.print("|\n");
System.out.print(   "| 61 to 100 |");
for (int i=0; i<  sixtyOrMore; i++)
{
   System.out.print("*");
}
System.out.print("|\n");
System.out.println("_____________");
```

Not quite what it will print out, but close

Review

- Arrays
  - Declaration
  - Length
  - Syntax
  - Examples