

COMP 1010- Summer 2015 (A01)

Jim (James) Young

young@cs.umanitoba.ca

jimyoung.ca



arrays!



Let's make a program:

A particle (point) shoots away from the mouse at a random speed in X and Y (== random angle)

If it hits the edge of the screen, start over with a new random speed

Let's update.. 3 points

Requires copy-paste variables

Repetition of code – can clean up a little with methods, but there is a limit to this

What if I want 50? 1000?

- not feasible

what is an array?

an array is simply an ordered **list of data** of a given type:

you can have an array of 100 **ints**

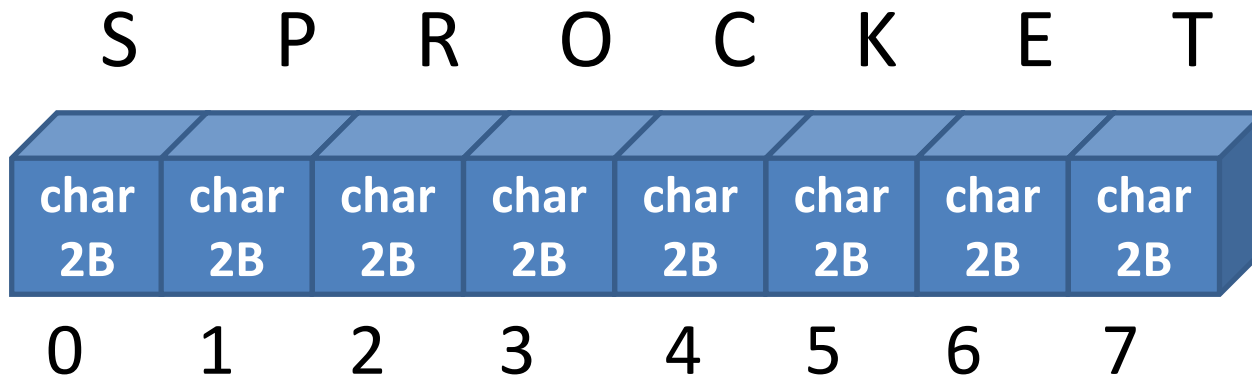
you can have an array of 10,000 **Strings**

you can have an array of 5 chars

Arrays are essential to programming and you will use them often!

we've actually seen the array before...

A Processing String, underneath, is an array of characters:



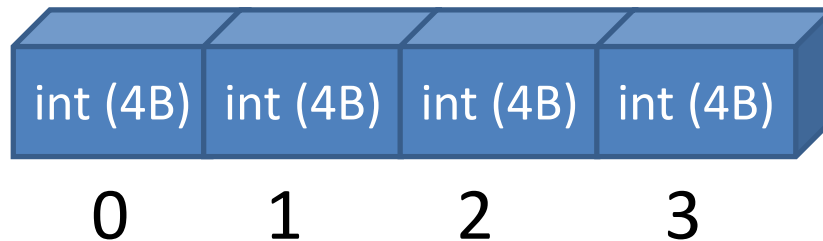
this is an array of characters, of length 8.

note that each box (item) in the array has its own index!! (`String.charAt(index)`)

the Array!

note: an array is a list of data of a given type. You can make an array of any data type!

here is an array of 4 integers:



as with strings, each box has its own unique index:

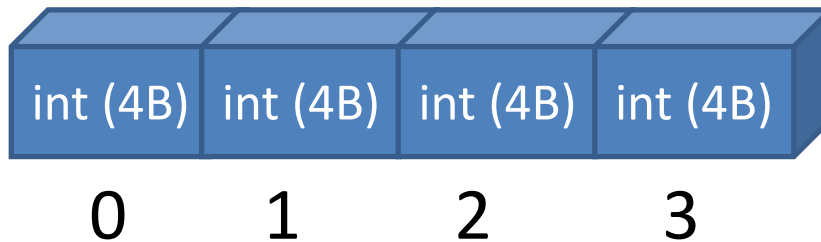
also as with strings, the first index is 0

Arrays have a fixed length

Array length cannot change!

You decide the length when you create an array

You cannot grow or shrink it



What if you need an array to grow or shrink?

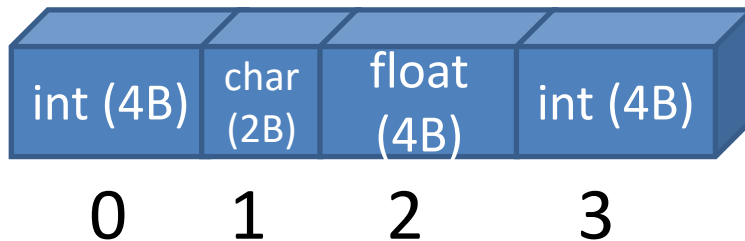
(classic final exam question)

Make a new one of the right size and copy your data

Arrays are homogenous

note: an array only has a single type, types cannot be mixed:

you cannot create an array like this:



choose your array type carefully because you are stuck with it!!

Syntax!

creating an array has two steps:

1. declare the variable container for the array
2. create the array (allocate memory) and assign it to the container.

1:

```
type [ ] variableName; // type is any Processing type.  
double [ ] studentScores;  
String [ ] studentNames;
```

note: square brackets: []. this marks an array

note: you do not tell how large (how many boxes..) your array is at this point!!

once variable is declared, have to
make the array...

2. create the array (allocate memory) and assign it to
the container

```
variableName = new type [arraySize];  
double[] studentScores;  
studentScores = new double[30];
```

combined **declaration** and **creation** is very common:

```
int[] studentAges = new int[30];
```

note: once an array is created it CANNOT be resized.
choose your size wisely!!

examples of **declaration** and **instantiation**

declaration

```
double [] gameScores  
String [] blogComments  
long [] bigNumbers
```

instantiation

```
= new double[5000];  
= new String[200];  
= new long[100];
```

note: must both **declare** the variable and **instantiate** the array.

using the array

```
int[] numbers = new int[10]; // array of 10 ints
```

you can access any bin in the array just by putting it in square brackets:

`numbers[0]` ← first int in the array

`numbers[5]` ← sixth int in the array

`numbers[9]` ← last...

`numbers[10]` ← ???

this will raise an error: index out of bounds

accessing array bins

store values:

```
variableName[bin] = value;  
studentScores[15] = 20;  
studentScores[0] = 15;
```

retrieve values

```
int i = studentScores[15];  
if (studentScores[i] > 90) message = "woohoo";
```

can be used anywhere a normal value can be used!

```
studentScores[15] = studentScores[14]*2;
```

array bins:

`double[] values = new values[100]; // 100 bins`
`values[3]` to access 4th bin (bin 3).



you can use a literal (e.g, 3, 4) or a variable as the bin index:

```
int i = 5;
```

```
values[i] to access bin i
```

the index **MUST BE AN INTEGER TYPE**

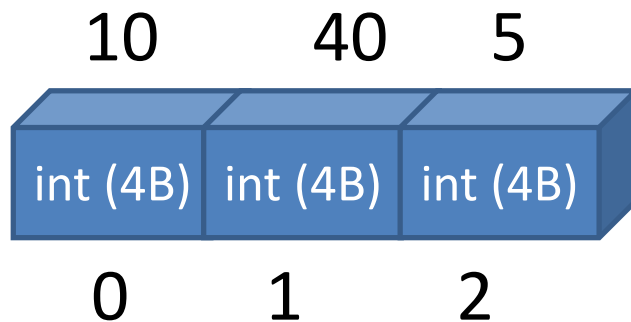
`values[5.5]` ← doesn't make sense

`values["yo!"]` ← doesn't make sense

`values[false]` ← doesn't make sense

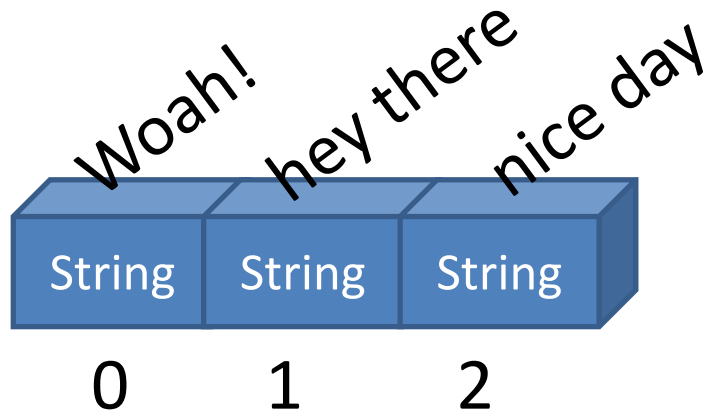
example:

make an array of integers of size three, and put three numbers into it: 10, 40, 5



Example cont...

make an array of Strings of size three, and put three messages into it: “woah!” “hey there” “nice day”



Example cont...

Use the arrays to print out the three messages at different locations

Good progress-

Only made one variable, not three

But still copy pasted code.. Any ideas?

Use a for loop to go through the array bins 0,1,2

Change sizes[0] to sizes[i]

Add two more lines

Two things:

- make the arrays bigger
- store the new data

And everything else still works.

Example:

Make arrays of size 8, one for x, one for y

- fill it with 8 random points at program start
- draw lines from the mouse to those points
- if the mouse is pressed, make new random points

Scale up to 1000;