COMP1917: 09 Arrays and Strings

Sim Mautner

s.mautner@unsw.edu.au

August 15, 2016

Arrays

```
int sum(int n1, int n2);

int sum(int n1, int n2, int n3);

int sum(int n1, int n2, int n3, int n4);

int sum(int n1, int n2, int n3, int n4, int n5);

int sum(int n1, int n2, int n3, int n4, int n5, int n6);

int sum(int n1, int n2, int n3, int n4, int n5, int n6, int n7
```

Arrays

```
// Declare an array with 10 elements
// and initialises all elements to 0.
int myArray[10] = {0};
```

	myArray
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0

Arrays

```c
// Declare an array with 10 elements
// and initialises all elements to 0.
int myArray[10] = {0};

// Put some values into the array.
myArray[0] = 3;
```

	myArray
0	3
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0

Arrays

```c
// Declare an array with 10 elements
// and initialises all elements to 0.
int myArray[10] = {0};

// Put some values into the array.
myArray[0] = 3;
myArray[5] = 17;
```

	myArray
0	3
1	0
2	0
3	0
4	0
5	17
6	0
7	0
8	0
9	0

Arrays

```c
// Declare an array with 10 elements
// and initialises all elements to 0.
int myArray[10] = {0};

// Put some values into the array.
myArray[0] = 3;
myArray[5] = 17;
myArray[10] = 42; // <-- Error
```

	myArray
0	3
1	0
2	0
3	0
4	0
5	17
6	0
7	0
8	0
9	0

Array Representation

- An array's name is a pointer to the first element of the array.
- Can be referred to as int num[] or int *num.

Arrays Exercises

- Ex 1: Write an application which creates an array with 15 numbers, reads in values and then prints out the array.
- Ex 2: Improve exercise 1 to use functions to load and print the array.
- Ex 3: Write a function (and an application to use it) which takes in an array of integers, and an integer, and returns the number of times that number occurs in the given array.

Multidimensional Arrays

- Arrays can have 2, 3 or more dimensions.
- Used to store data for easier access.
- Eg game boards (chess, checkers), data that's easier accessed via more than one dimension.
- Tic-Tac-Toe demo

Strings

- Text
- Hard-coded in double quotes (the same way a `char` is hard-coded in single quotes)
- Stored in memory as an array of characters
- Ensure enough space when creating the variable
- Has a null character ('\0') at the end to indicate the end of the string
- `%s` is the corresponding placeholder
- `scanf` doesn't require an ampersand

```
scanf("%s", myString);
```

- Reading a string using scanf, results in everything up to the first whitespace being read in

String Exercises

- Ex 1: Write a function which takes in a string and returns the number of characters in the string.

```
int getStringLength(char * string);
```

- Ex 2: Write a function which copies one string into a second array of characters.

```
int copyString(char * destination, char * source);
```

String Libraries

```
#include <string.h>
```

- `man string` - execute on the terminal to get a list of functions available
- `man strlen` - execute on the terminal to get details on how to use the `strlen` function

- `strlen` - gets the length of a string
- `strcmp` - compares two strings, returns 0 if they're the same, negative number if they're in alphabetical (ascii) order, positive number if they're out of order
- `strcasecmp` - same as `strcmp` except ignoring case

`fgets` - Read a Whole Line

- `fgets` reads in a whole line of text, up to and including a newline character.
- Parameters:
  1. String - the string to read into.
  2. Size - the maximum number of characters to read in. (Don't forget to leave room for the null character.)
  3. File - Where to read the string from. To read from the keyboard, use the keyword `stdin`
- Always includes the '\n' character at the end of the string.
- To remove the newline character:

```
string[strlen(string)-1] = '\0';
```

- Ex 3: Write an application which uses fgets to read in a line of text.

Commandline Arguments

- `argc` - the number of arguments (or parameters) passed into the application
- `argv` - an array of strings, each string is an argument to the application

- Ex 4: Write an application which prints out the commandline arguments it receives.