VIRTUAL COURSE
BUILD YOUR OWN DATA LOGGER

WILDLABS.NET
[ The conservation technology network ]

FREAKLABS
MODULE 3 - SUBMODULE 1 - LAB 2
SETTING AND ACCESSING PARAMETERS
Command Line - Lab 1 Recap

We:

- Installed cmdArduino library
- Went through the library functions including cmd.begin(), cmd.add(), cmd.poll(), cmdFunctionName()
- Wrote our first command: Hello Command Line
Command Line - Lab 2

Goal:

- Understand how the cmdFunctionName() takes in and handles arguments
- Write a command that lists its own arguments
- Understand how to change parameters by passing in arguments through the serial monitor
- Set the date on a pretend real time clock, and print it out, all via the serial console

Recap

- Parameters (similar to variable) = refers to an unknown data value capable of being set by the user, eg. Through the command line.
- Arguments = a value that is passed into a command, function, or routine
Data types (in C++)

- Character (char) is a data type used to store a single character which might be a letter, a space, a number or a symbol. Eg. “a” “1” “!”

- String (str) = a series of characters. Can contain letters, spaces, numbers and symbols. eg. “Hello World!”, “I’m number 1”.

- Integer (int) is a number with no decimal point. Can be positive, negative or zero. eg. 1, 34, -1024, 0

- Floating point number (float) is a number with decimal point that can move. eg 34.4, 1.15, 234.02567,
Arrays

- An array is a group of related data values called elements.
- The array elements (or values) must be the same data type.
- When declaring an array we specify the data type, arrayName and arrayLength.

Eg. type arrayName[arrayLength] = [..., ..., ...]

- Access elements (or values) in the array via the element’s index (or position in the array).
- Index count starts from 0

Eg. arrayName[index]
Array

If we type `list hello command line` in the serial monitor:

```java
char args[4] = ["list", "hello", "command", "line"]
```

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>“list”</td>
<td>“hello”</td>
<td>“command”</td>
<td>“line”</td>
</tr>
</tbody>
</table>

`args[0] = “list”`

`args [1] = “hello“`

`args [2] = “command”`

`args [3] = “line”`
Programming Refresh – “for” Loop

“for” Loop

- A “for” Loop is used to repeat a specific block of code a certain number of times.
- Often used to loop through an array, and do something to all the elements.

Eg.

```cpp
int i;
for (i=0; i<arrayLength; i++)
{
    Serial.println(arrayName[i]);
}
```
void cmdFunctionName(int argCnt, char **args)

int argCnt

• The number of arguments we’re passing to our function through our the serial monitor
• Gives us the array length for args
• Defaults to 1 which is the command keyword (or first word we type)

char **args

• an array of strings called args.
• Takes each word we’ve typed as an element of the args array
eg. If we type list hello command line into the serial console it creates:
string args[“list”, “hello”, “command”, “line”]
cmdArduino – cmd.conv(arg1)

What we type into the serial console is a string (ASCII text).

Therefore we need to convert any numbers from a string into an integer using the `cmd.conv(args[index])` function

`cmd.conv(args[index])`

If we type `list 5` into the serial console, and want to convert “5” into a number 5, we write:

```c
int variableName = cmd.conv(args[1])
```

Now let’s get into the code!

Writing a command that lists its own arguments

`cmdList`

Setting a date on our pretend realtime clock (changing the date through the serial console)

`cmdSetDate`
Troubleshooting!

If your code doesn’t verify check:

• The correct board is selected
• The correct port is selected
• There’s no typos in your code
• Function and variables names are consistent (case sensitive)
• Each opening bracket has a closing bracket
• Each statement has a semi-colon at the end
COMING UP - MODULE 3 - SUBMODULE 1

LAB 3: CHANGING PARAMETERS IN REAL TIME