

# Introduction to IoT

School Year 2023-2024

Valsalice


# Course Structure

1	Introduction and Basics	
2	Basic Data Types and Operators	
3	Control Structures Pt. 1	
4	Control Structures Pt. 2	
5	Functions and Scope	
6	Arrays	
10	Advanced String Usage	DEC 19
11	Custom Data Types	JAN 9

7	Introduction to Contiki-NG and nRF52840	
8	Sensing and Actuating with Contiki-NG	
9	Timers and Concurrency	DEC 12
12	Basic Communication and Networking	JAN 16
13	Introduction to RPL and Network Routing	JAN 23
14	Advanced Protocols: TSCH and 6TiSCH	JAN 30
15	Advanced Topics in Wireless Communication	FEB 6
16	Reliable Data Transfer Challenge	FEB 20 FEB 27 MAR 5

■ = Core Topics    ■ = Optional Topics

# Open your Virtual Machines

1. Turn on your Laptops
2. Login to Windows using "User"
3. Open the **Virtual Box** program
4. Add a new Virtual Machine (**Ctrl + A**)
5. Open the **VirtualBox** folder  (**NOT** the .VirtualBox)
6. Select the **nRF52840LAB** file
7. Click **Start**

# Prepare the Coding Environment

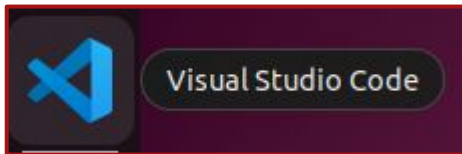
1 Start the Virtual Machine **nRF52840LAB**

2 Log-in using credentials:

Username: **ubuntu**

Password: **ubuntu**


3 Open **Visual Studio Code** (use the App bar on the left)

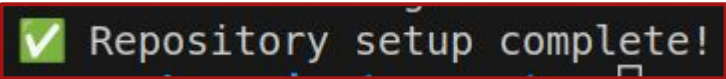


# Prepare the Coding Environment

- 4 From the Terminal:

```
make setup
```

- 5 

- 6 



If you see **any (yellow) errors** input the credentials again

- 7 Open the **week10** folder in the terminal

Right click on the left + **“Open in Integrated terminal”**



# Recap: Data Types

C has a number of primitive data types:

**int**

42

1200

1\_200

-3

**float**

3.14

0.00001

-2.1

**char**

'A'

'@'

'\n'

**bool**

true

false

Strings are *NOT* a primitive data type, and have special syntax.

**strings**

"Hello"

"A"

"I am a full sentence!"

# Recap: Variables

A variable is a named container that stores data or values.

```
int x = 42;  
float y = -0.12;  
char w = 'A';  
char z[50] = "Full sentence";
```

Booleans require a custom include statement:

```
#include <stdbool.h>  
bool hello = true;
```

# Recap: Boolean Operators

Greater than	>
Greater or equal than	>=
Less than	<
Less or equal than	<=
Equals	==
Not equals	!=
Not	!



# Recap: Chaining Comparisons

- **and** (both must be true)

```
true && false
```

```
(5 < 6) && (5 < 10)
```

- **or** (either must be true)

```
true || false
```

```
(5 < 3) || (5 < 10)
```

- **not** (negation)

```
!true
```

```
!(5 < 3)
```

# Recap: If-Statement chaining

You can chain multiple conditions with **else if**.

What is the difference between these two snippets of code?

```
int num;
scanf("%d", &num);

if (num < 3) {
    printf("Small number\n");
} else if (num < 10) {
    printf("Medium number\n");
}
```

```
int num;
scanf("%d", &num);

if (num < 3) {
    printf("Small number\n");
}
if (num < 10) {
    printf("Medium number\n");
}
```

# Recap: While-Loops

Repeat parts of  
your code!

```
int num;  
printf("Input a number greater than 100: ");  
scanf("%d", &num);  
  
while (num <= 100) {  
    printf("Wrong number, try again: ");  
    scanf("%d", &num);  
}  
  
printf("Well done!\n");
```



# Recap: For-Loops

Repeat a **specific** amount of times!

```
int x;  
  
for (x = 1; x <= 5; x++) {  
    printf("Hello %d\n", x);  
}
```

```
int x = 0;  
  
while (x < 5) {  
    x += 1;  
    printf("Hello %d\n", x);  
}
```

# Recap: Array Elements

Modifiable containers for data.

To access array elements you can use the **[index]** operator.

**NOTE:** List indices start from **0**

index:	0	1	2	3	4
	17	28	33	56	6

```
int array[] = {17, 28, 33, 56, 6};
```

```
printf("%d\n", array[0]);
```

```
printf("%d\n", array[3]);
```

# Recap: Assigning Array Elements

To assign array elements you can use the **[index]** operator on the left-hand-side of a statement (like a variable)

```
int array[] = {17, 28, 33, 56, 6};  
array[3] = 100;  
array[2] = -7;
```

```
printf("%d\n", array[0]);
```

```
printf("%d\n", array[3]);
```



# Recap: Functions

Functions are custom snippets of reusable code:

1. If the **return type** is **void** the function does NOT return.
2. If the **return type** is NOT void, it MUST use **return**.

1

```
// Function to print a number
void print_num(int num) {
    printf("%d\n", num);
}
```

2

```
// Function to add two numbers
int add(int num1, int num2) {
    return num1 + num2;
}
```

# Recap: Calling Functions

Functions can be called any number of times:

```
// Function to add two numbers
int add(int num1, int num2) {
    return num1 + num2;
}
```

```
int x = add(4, 100);
int y = add(60, 30);
int z = add(x, y);
```



# The String Library

```
/* Calculates the length of a given string */  
size_t strlen(char* str);  
  
/* Compares strings, returns 0 when equal, 1 (or -1) otherwise */  
int strcmp(char* first_str, char* second_str);  
  
/* Append a copy of the src string to the end of dest string */  
int strcat(char* dest, char* src);
```

# Exercise

Implement all functions inside (**simple.c**).

Function descriptions are the specification

**To run automated tests:** `make simple.test`

# Save remotely your Changes

1

```
make save
```

2

```
|Password
```

```
Git: https://aspina@git.spina.me (Press 'Enter' to confirm or  
'Escape' to cancel)
```

3

```
✓ Changes committed and pushed. All done!
```

# Strings as Arrays

Strings are nothing more than arrays of characters:

```
char string1[] = "Hello";  
char string2[] = {'H', 'e', 'l', 'l', 'o'};
```

Index:

	0	1	2	3	4
<code>char string2[] = {'H', 'e', 'l', 'l', 'o'};</code>					

```
printf("%c\n", string2[0]);
```

```
printf("%c\n", string1[3]);
```

# More String Library Functions

```
/* Convert the character to lower-case */  
char tolower(char character);  
  
/* Convert the character to upper-case */  
char toupper(char character);
```

# Exercise

Implement all functions inside (**advanced.c**).

Function descriptions are the specification

**To run automated tests:** `make advanced.test`

# Save remotely your Changes

1

```
make save
```

2

```
|Password
```

```
Git: https://aspina@git.spina.me (Press 'Enter' to confirm or  
'Escape' to cancel)
```

3

```
✓ Changes committed and pushed. All done!
```

# Trivia Time!



The image shows a presentation slide for a Christmas Trivia Quiz. The slide has a dark red background with a subtle snowflake pattern. In the top right corner, there is a yellow and red striped Christmas ornament icon and the AhaSlides logo. The main title "Christmas Trivia Quiz" is centered in white text. Below it, the URL "ahaslides.com/CI5KK" is displayed in a large, bold, white font. In the bottom center, there is a cartoon illustration of Santa Claus next to a red gift box with a yellow bow. The bottom right corner shows a user count of "0/7". On the left side, there is a dark blue sidebar containing a QR code, the text "Join at: ahaslides.com/CI5KK", and a "Menu" button at the bottom.

Christmas Trivia Quiz

[ahaslides.com/CI5KK](https://ahaslides.com/CI5KK)

Join at:  
[ahaslides.com/CI5KK](https://ahaslides.com/CI5KK)

Menu

0/7



# End of Class

See you all next week!