Anna - Giuseppe

First step: We created a new TinyOs project called challenge3 in which we created three files called Challenge3AppC, Challenge3C and make file.

```
module Challenge3C{
  uses interface Boot;
  uses interface Timer<TMilli>;
  uses interface Leds;
  }
```

Second step: We implemented the main module in which we used a Boot interface where we initialized the timer to 60000 ms, we switched off the three LEDs and set the status variable. Inside the module we also added a Timer and LED interfaces.

Third step: In the implementation we initialized a variable called personal code and other working variables. Inside the event Timer.fired() we implemented the ternary conversion steps and we changed the state of the LED with the toggle function, according to the value of the variable rem (remainder).
Then we sent all the led status with the function printf and at the end of the conversion we switched off all the LEDs.

Fourth step: We wrote the Challenge3AppC according to the previous file. We implemented the make file including the flag useful for the printf.

Fifth step: We exploit the simulator Cooja to run the built project using a sky mote with a serial socket (server) with the purpose of sending data to node-red.

Sixth step: In node-red we used an input TCP to get the data from Cooja. We processed the data with a function node and sent them to our public channel previously setted using mqtt connection. We implemented a delay node to be sure to avoid a DDOS to ThingSpeak server (even if we set a specific timer in TinyOs).