```cpp
#include "UserCode.hpp"
#include "UtilityFunctions.hpp"
#include "Vec3f.hpp"

#include <stdio.h> //for printf

//An example of a variable that persists beyond the function call.
float exampleVariable_float = 0.0f;  //Note the trailing 'f' in the number. This is to force single precision floating point.
Vec3f exampleVariable_Vec3f = Vec3f(0, 0, 0);
int exampleVariable_int = 0;

//We keep the last inputs and outputs around for debugging:
MainLoopInput lastMainLoopInputs;
MainLoopOutput lastMainLoopOutputs;

//Some constants that we may use:
const float mass = 30e-3f;  // mass of the quadcopter [kg]
const float gravity = 9.81f;  // acceleration of gravity [m/s^2]
const float inertia_xx = 16e-6f;  //MMOI about x axis [kg.m^2]
const float inertia_yy = inertia_xx;  //MMOI about y axis [kg.m^2]
const float inertia_zz = 29e-6f;  //MMOI about z axis [kg.m^2]

const float dt = 1.0f / 500.0f; //[s] period between successive calls to MainLoop

MainLoopOutput MainLoop(MainLoopInput const &in) {
  //Your code goes here!
  // The function input (named "in") is a struct of type
  // "MainLoopInput". You can understand what values it
  // contains by going to its definition (click on "MainLoopInput",
  // and then hit <F3> -- this should take you to the definition).
  // For example, "in.joystickInput.buttonBlue" is true if the
  // joystick's blue button is pushed, false otherwise.

  //Define the output numbers (in the struct outVals):
  MainLoopOutput outVals;
//  motorCommand1 -> located at body +x +y
//  motorCommand2 -> located at body +x -y
//  motorCommand3 -> located at body -x -y
//  motorCommand4 -> located at body -x +y

  //  If the blue button is pressed, we set all motor outputs to 50
  if (in.joystickInput.buttonBlue) {
    outVals.motorCommand1 = 50;
    outVals.motorCommand2 = 50;
    outVals.motorCommand3 = 50;
    outVals.motorCommand4 = 50;
  }
  //  Otherwise, we want to set the all motor outputs to 0
  else {
    outVals.motorCommand1 = 0;
    outVals.motorCommand2 = 0;
    outVals.motorCommand3 = 0;
    outVals.motorCommand4 = 0;
  }



  //copy the inputs and outputs:
  lastMainLoopInputs = in;
  lastMainLoopOutputs = outVals;
  return outVals;
}

void PrintStatus() {
  //For a quick reference on the printf function, see: http://www.cplusplus.com/reference/cstdio/printf/
  // Note that \n is a "new line" character.
  // Also, note that to print a `float` variable, you have to explicitly cast it to
  //  `double` in the printf function, and explicitly specify precision using something
  //  like %6.3f (six significant digits, three after the period). Example:
  //   printf("  exampleVariable_float = %6.3f\n", double(exampleVariable_float));

  //Accelerometer measurement
  printf("Acc: ");
  printf("x=%6.3f, ",
         double(lastMainLoopInputs.imuMeasurement.accelerometer.x));
  printf("\n");  //new line
  printf("Gyro: ");
  printf("x=%6.3f, ", double(lastMainLoopInputs.imuMeasurement.rateGyro.x));
  printf("\n");  //new line

  printf("Example variable values:\n");
  printf("  exampleVariable_int = %d\n", exampleVariable_int);
  //Note that it is somewhat annoying to print float variables.
  //  We need to cast the variable as double, and we need to specify
  //  the number of digits we want (if you used simply "%f", it would
  //  truncate to an integer.
  //  Here, we print 6 digits, with three digits after the period.
  printf("  exampleVariable_float = %6.3f\n", double(exampleVariable_float));

  //We print the Vec3f by printing it's three components independently:
```

```cpp
  printf("  exampleVariable_Vec3f = (%6.3f, %6.3f, %6.3f)\n",
         double(exampleVariable_Vec3f.x), double(exampleVariable_Vec3f.y),
         double(exampleVariable_Vec3f.z));

  //just an example of how we would inspect the last main loop inputs and outputs:
  printf("Last main loop inputs:\n");
  printf("  batt voltage = %6.3f\n",
         double(lastMainLoopInputs.batteryVoltage.value));
  printf("  JS buttons: ");
  if (lastMainLoopInputs.joystickInput.buttonRed)
    printf("buttonRed ");
  if (lastMainLoopInputs.joystickInput.buttonGreen)
    printf("buttonGreen ");
  if (lastMainLoopInputs.joystickInput.buttonBlue)
    printf("buttonBlue ");
  if (lastMainLoopInputs.joystickInput.buttonYellow)
    printf("buttonYellow ");
  if (lastMainLoopInputs.joystickInput.buttonStart)
    printf("buttonStart ");
  if (lastMainLoopInputs.joystickInput.buttonSelect)
    printf("buttonSelect ");
  printf("\n");
  printf("Last main loop outputs:\n");
  printf("  motor command 1 = %6.3f\n",
         double(lastMainLoopOutputs.motorCommand1));
}
```