```
#include "pico/stdlib.h"
#include "hardware/i2c.h"
#include <stdio.h>

// --- BH1750 Definitions ---
#define BH1750_ADDR 0x23 // Default I2C address for BH1750 sensor
#define BH1750_CONT_H_RES_MODE 0x10

static i2c_inst_t *bh1750_i2c;
static uint8_t bh1750_address = BH1750_ADDR;

// --- BH1750 Initialization ---
void bh1750_init(i2c_inst_t *i2c, uint sda, uint scl) {
    bh1750_i2c = i2c;

    // Initialize I2C at 100 kHz
    i2c_init(bh1750_i2c, 100 * 1000);

    // Set up I2C pins
    gpio_set_function(sda, GPIO_FUNC_I2C);
    gpio_set_function(scl, GPIO_FUNC_I2C);
    gpio_pull_up(sda);
    gpio_pull_up(scl);

    // Send power on and continuous high resolution mode command
    uint8_t cmd = BH1750_CONT_H_RES_MODE;
    i2c_write_blocking(bh1750_i2c, bh1750_address, &cmd, 1, false);
    sleep_ms(180); // Wait for first measurement
```

```
}

// --- Read Light Level in Lux ---
float bh1750_read_light_level() {
    uint8_t data[2];

    // Read 2 bytes from the sensor
    int result = i2c_read_blocking(bh1750_i2c, bh1750_address, data, 2, false);
    if (result != 2) return -1.0f;

    uint16_t raw = (data[0] << 8) | data[1];
    return raw / 1.2f; // Convert to lux
}

// --- Main Program ---
int main() {
    stdio_init_all(); // For USB serial output

    // Initialize BH1750 sensor using I2C1 on GPIO2 (SDA) and GPIO3 (SCL)
    bh1750_init(i2c1, 2, 3);

    while (true) {
        float lux = bh1750_read_light_level();

        if (lux >= 0) {
            printf("Light Level: %.2f lux\n", lux);
            if (lux > 10) {
                printf("Light Detected\n");
```

```
            } else {
                printf("Light Not Detected\n");
            }
        } else {
            printf("BH1750 read failed.\n");
        }

        sleep_ms(1000); // Delay 1 second
    }

    return 0;
}
```