

Programming onboarding task

This task is designed to check your coding ability and problem-solving skills whilst also getting you familiar with our method of operation. In this task, you will be required to write a simple program (or module) that meets the specifications outlined below. This should be relatively easy so try not to use AI tools to complete it! If you're stuck, try to search online for help before trying to complete it with AI.

Task Overview

You are to write a function that accepts a pressure reading from a sensor in Pascals (Pa). If the pressure is between 10 000 and 110 000 Pa (inclusive), the function should return a `STATUS_OK` code. If the pressure is outside this range, or if the number is mathematically invalid (e.g., NaN, infinity), return a `STATUS_ERROR` code and increment a global `fault_count`.

Guidelines

- Use the provided `status_defines.h` file for return codes.
- Implement the function in `range_check.c`.
- You do not need to modify `main_test.c`, but you may read through it to understand how your function will be tested. Try to avoid hardcoding values in your implementation and provide a general solution.

Running your code

1. Ensure that you have a C compiler installed on your machine.
2. Place all of the code files the **same directory** (this shouldn't be an issue if you download the whole task from the repository).
3. Open your terminal or command prompt and navigate to the folder where you placed the files and run the following command to compile the code:

```
gcc main_test.c range_check.c -o onboarding_task
```

4. You now just need to run the compiled program! This can be done by executing the following command in your terminal:

```
./onboarding_task
```

Evaluation

The program will output a score out of 3. A successful implementation should be able to achieve a perfect score of 3/3 and will demonstrate your understanding of:

- Pointer-based memory updates (for the fault count)
- Input validation (range checking)
- Mathematical safety (handling NaN and infinity)