# Experiment 2: helloworld

**Coding is needed. But no code submission is required.**

**Convert to PDF before uploading.**

Related project description:

https://fxlin.github.io/p3-tee/quickerstart/

https://fxlin.github.io/p3-tee/helloworld/

## Context & Sessions (20%)

Use 1-2 sentences for each question below.

1. What is a TEE "context"? Explain the purpose of this object.

The TEE Context are containers for sessions and their purpose is to create a logical connection between the client and the TEE.

1. In one CA, how many TEE contexts can be simultaneously alive (i.e. initialized but not finalized)? You can determine this with a small experiment, e.g. keep creating TEE contexts until the creation fails.

It seems like you can only have 1022 TEE context alive at a time.

1. In one context, how many sessions can be opened simultaneously? You can determine this with a small experiment, e.g. keep creating TEE sessions in the same context until the creation fails.

Only 17 sessions can be created per one session.

You may need to write some experiment code to validate your answers above. Just state your observations.

## Change helloworld (40%)

The code description: https://fxlin.github.io/p3-tee/helloworld/#app-1-helloworld

Change the source code: if the resultant value (i.e. after incrementing or decrementing) is NOT in the range of [0, 100], TA should return an error to the CA.

How did you implement the above logic? In space below, paste the **key** lines you added/modified, which should not be more than 20 lines (10%)

A grey background with white text

Description automatically generated

Implemented in static TEE\_Result inc\_value(uint32\_t, param\_types, TEE\_Param params[4]) and static TEE\_Result dec\_value(uint32\_t param\_types, TEE\_Param params[4]) in ta/hello\_world\_ta.c.



Implemented in int main(void) in host/main.c.

How do you verify that your code works properly? E.g. what values does CA send to TA? Does your CA print an error code returned by the TA? Attach a screenshot to prove it. (30%)

The CA will send parameters to the TA and the TA will return the modified parameters with an operation code. The CA will print an error code returned by the TA regarding out-of-bounds issue.

Test Cases:

**In-bounds:**



Increment 42 to 43 is successful because 43 is within the bounds

A black screen with white text

Description automatically generated

Decrement 42 to 41 because 41 is within the bounds

**Out-of-bounds:**

****

Incrementing 42 to 142 results in an out-of-bounds error because 142 is greater than 100.



Decrementing 42 to -58 results in an out-of-bounds error -58 is less than 0.

**Edge Cases:**



Incrementing 100 to 101 results in an out-of-bounds error because 101 is greater than 100.

A black screen with white text

Description automatically generated

Incrementing -1 to 0 results in successful execution because 0 is in the bounds.



Decrementing 0 to –1 results in an out-of-bounds error because -1 is less than 0.

A black screen with white text

Description automatically generated

Decrementing 101 to 100 results in a successful execution because 100 is in the bounds.