

1 Introduction

This section explains the implementation of your test file. There will be five test cases. The test file will begin with the following:

```
import studentLibrary as sl
import justinLibrary as jl
import time
```

All tests of time will be calculated using the following:

```
start = time.perf_counter()
#code to run
end = time.perf_counter()
```

Where *start* and *end* are the time in seconds to the highest precision your system can handle. Only the difference between them is meaningful. Each separate function should have the time output.

2 Test Cases

1. Find the time it takes to run the calculation of `sl.sin(x)` and `jl.sin(x)` for every value in the range `[0, 0.001, 0.002, ..., 1]`.
2. Find the time it takes to calculate the standard deviation of the array found in `trial_array.txt` (10,000 values) using both `sl.st_dev()` and `jl.st_dev()`
3. Find the time it takes to run `sl.root(x)`, `jl.root(x)`, `sl.exp(x, 0.5)`, and `jl.exp(x,0.5)` 1,000 times on the same value. These should all provide the same output, so why does each one take different amounts of time?
4. Using numpy, set the seed to 42, and run “`samples = np.random.normal(loc=0, scale=1, size=10000)`” then, find run `sl.` and `jl.mean()`, `.median()`, and `.std_dev()`. As this distribution is normal, these should output the expected values of 0, 0, and 1.
5. Run a loop of 1000 times which does the following

```
x=27.01
for i in range(1000):
    x = sl.exp(x)
    x = sl.log(x)
```

What is the output, and is it what you expected from running this calculation?

6. Create your own test case, and explain why this illustrates something not seen in the other cases.