

Note the issue

- ◆ Hysteresis — memory of what came before
 - ◆ A callable object can be stateful
- ◆ This may be desirable to optimize performance
- ◆ It may be confusing if done badly

We generally expect idempotence:
functions do the same thing each time

Exercise: Callable

- ◆ Let's revise the **eng()** function from long back

```
class Engine:
```

```
    def __init__(self):
```

```
    def __call__(self, tach: float) -> int:
```

- ◆ Recall:

```
n100 = lambda x: int(round(x, -2))
```

```
eng = lambda r: n100(0.7724*n100(r)**1.0134)
```