Tawan Boonma
63105454272

HOF_maxer.py

Completed 100% of this problem from doctests. By print the value from the function received from smoke first, and compare with the new value from function fire by compute them in function haze and return function fire with max value.

```
def square(x):
    return x * x

def maxer(smoke):
    """Return a repeatable function fire(y) that prints the largest smoke(y) so
    >>> g = maxer(square)
    >>> h = g(2)(1)(3)(2)(-4) # print the largest square(y) so far
    4
    4
    9
    9
    16
    >>> h = maxer(abs)(2)(1)(3)(2)(-4) # print the largest abs(y) so far
    2
    2
    3
    3
    4
    """
    def fire(y):
        # fill one line of code here
        print(smoke(y))
        def haze(z):
            # fill an if condition and a line of code in the if block here
            if smoke(y) > smoke(z):
                z = y
            # fill a return line here
            return fire(z)
        return haze
    return fire
```

```
/Library/Frameworks/Python.framework/Versions/3.9/bin/python3
❯ /Library/Frameworks/Python.framework/Versions/3.9/bin/python3
Python 3.9.1 (v3.9.1:1e5d33e9b9, Dec  7 2020, 12:10:52)
[Clang 6.0 (clang-600.0.57)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> def square(x):
...     return x * x
...
>>> def maxer(smoke):
...     def fire(y):
...         # fill one line of code here
...         print(smoke(y))
...         def haze(z):
...             # fill an if condition and a line of code in the if block here
...             if smoke(y) > smoke(z):
...                 z = y
...             # fill a return line here
...             return fire(z)
...         return haze
...     return fire
...
>>> g = maxer(square)
>>> h = g(2)(1)(3)(2)(-4)
4
4
9
9
16
>>> h = maxer(abs)(2)(1)(3)(2)(-4)
2
2
3
3
4
>>>
```