

Zadanie G

Instancje

Zaimplementuj dekorator `inst`, którego zadaniem będzie zliczanie wszystkich utworzonych instancji obiektów udekorowanej klasy finalnej. Udekorowanie klasy finalnej `A` dekoratorem `@inst` ma umożliwiać wywołanie instrukcji `A.num_of_inst()`, która poda liczbę wszystkich stworzonych (również już tych odśmieconych) instancji klasy `A`.

Wysyłanie rozwiązania

Należy wysłać jeden plik o nazwie `inst.py`.

Dostępna pamięć: **32MB**

Przykład

Dla testu:

```
from inst import inst
```

```
@inst
```

```
class A:
```

```
    def __init__(self, name, job="none"): show ('ok',) {}
```

```
        self.name = name
```

```
        self.job = job
```

```
    def show(self,*args,**kwargs):
```

```
        print("show", args, kwargs)
```

```
    def __str__(self):
```

```
        return "Person: {}, job: {}".format(self.name, self.job)
```

```
a = A("a", job="manager")
```

```
aa = A("aa")
```

```
print(a)
```

```
print(aa)
```

```
A.num_of_inst()
```

```
a.show("ok", ok="OK")
```

```
A.show(a, "ok")
```

```
class C:
```

```
    def f(self):
```

```
        print(self.a)
```

```
@inst
```

```
class D(C):
```

```
    def __init__(self, a):
```

```
        self.a = a
```

```
    def __str__(self):
```

```
        return str(self.a)
```

```
d = D("d")
```

```
d = D("h")
```

```
d.f()
```

```
D.f(d)
```

```
print(d)
```

```
D.num_of_inst()
```

Poprawną odpowiedzią jest:

```
Person: a, job: manager
```

```
Person: aa, job: none
```

```
Liczba instancji klasy A: 2
```

```
show ('ok',) {'ok': 'OK'}
```

```
h
```

```
h
```

```
h
```

```
Liczba instancji klasy D: 2
```