Snake

用hex editor打开之后发现有pyc字样，考虑反编译pyc。使用uncompyle工具反编译pyc，得到py文件。 可以看到开头有一大堆类似于这样的定义。

```
fb4a1d7032971c72d8841c45ad4c85c2 = 22
f4d0ef13ca6a63eb31c9b6e64d7c10ec = 119
b820c751e796b8d57748580c2f0dea7f = 9
aaa08ed33947e41cdb67469de17a5304 = 53
b6c3bdb6702185d18efd549ece852869 = 87
e6337a65be0e81142404a700385653d3 = 19
af0c9bf91ee73245b65d18edc528be78 = int
fd7d29d26436c42ed939d6fcbe91a269 = 59
a62c60ecc4a6a10bb88b8212d2ef8444 = delattr
b2ebf8dbb9452dbbe91b5fc83c24f96c = str
# ......
```

之后是一段判断密码的代码，类似于下面这样：

```
def e40dd26a431281630458f650f71d929e(f5db77933b33ed118f6c87618bb2cafe):
    if b7b09ab3394c453125bb11ad680bdceb(f5db77933b33ed118f6c87618bb2cafe) !=
c09c4bdfc03e4d576f4e69a71cff88f1:
        return e144ac4cd2dc71dd7c4d6906ec407fce
    elif f5db77933b33ed118f6c87618bb2cafe[d56c1d827fb0817f7b1bf16e937b5439] !=
c7e65107d6af5499d3f2e44345d66273(cdc3e8ea764103585e63d587f96543de):
        return e144ac4cd2dc71dd7c4d6906ec407fce
    elif f5db77933b33ed118f6c87618bb2cafe[a2b941f886b9ebf394a6322a9a98f2f3] !=
c7e65107d6af5499d3f2e44345d66273(e5378990e30293b2bc1a9bf676474e2f):
```

我们可以将这段代码中的一些杂乱的变量用上面的定义替换，然后发现这段代码的作用，是逐位判断密码的正确性，由此我们可以根据ASCII码表得到最终的flag。