به نام خدا

این سوال متشکل از دو تابع با اهمیت است که به شرح زیر هستند:

modular_exponentiation: این تابع سه ورودی base , exponent , module را دریافت میکند . محاسبات مورد نظر از جمله چک کردن اول بودن exponent را به عهده میگیرد.

```
def modular_exponentiation(base, exponent, modulus):
    result = 1
    base = base % modulus
    while exponent > 0:
        if exponent % 2 == 1:
            result = (result * base) % modulus
        exponent = exponent // 2
        base = (base * base) % modulus
    return result
```

diffie_hellman:این تابع متشکل از چهار پارامتر g,p,b,a است که در آن مقادیر g و p که مقادیر ثابتی برای توان ژنراتور و ماژول می‌باشند و همچنین مقادیر a و b که توان‌هایی برای هر طرف هستند. این تابع ابتدا مقادیر a و b را محاسبه می‌کند که به ترتیب مقادیر g به توان a و b ماجول p هستند. سپس از این مقادیر برای محاسبه کلید مشترک استفاده می‌شود. در نهایت، این کلید مشترک را برمی‌گرداند.

```
def diffie_hellman(g, p, a, b):
    A = modular_exponentiation(g, a, p)
    B = modular_exponentiation(g, b, p)

    s1 = modular_exponentiation(B, a, p)
    s2 = modular_exponentiation(A, b, p)

    if s1 == s2:
        return s1
    else:
        return "Error: Key exchange failed."
```

و در نهایت پارامتر ها از طریق توابع کال میشوند و session key چاپ میشود:

به نام خدا

```
g = 2
p =
241031242692103258855207602219756607485695054850245994265411694195810883168261222889009385826134161467322714147790401219650364895705058263194273070680500922306273474534107340669624601458936165977404102716924945320037872943417032584377865919814376319377685986952408894019557734611984354530154704374720774996976375008430892633929555996888245787241299381012913029459299994792636526405928464720973038494721168143446471443848852094012745984428885933652689632091963391
9

a =
11010010100192031923123123124352342341312354576867565544342323253656453423232
43
b = 234342343247398472938479283749283749827398473984798 2

s = diffie_hellman(g, p, a, b)
print("Shared secret key:", s)
```

پایان