(1)

```python
N = 44426601460658291157725536008128017297890787463719427903128118036605 7
def fast_power(a,n):
    # print "%s %s" % (a, n)
    result = 1
    value = a
    power = n
    while power > 0:
        if power % 2 == 1:
            result = result * value % N
        value = value * value % N
        power = power/2
    return result

c = 0
c_last = 2
for i in range(2, 10001):
    c = fast_power(c_last, i)
    c = c % N
    c_last = c
print c
```

answer :

588125490264460931874486873686811669774136875258579427457988006083 69

(2) 2^{1024} **are all 6-smooth，但是**2^{1024} **不整除6!**

(3)

```python
N = 86202154764363158239699821220872291428825864423479130795058291644274722203979560941774193227831712 1
def fast_power(a,n):
    # print "%s %s" % (a, n)
    result = 1
    value = a
    power = n
    while power > 0:
        if power % 2 == 1:
            result = result * value % N
        value = value * value % N
        power = power/2
    return result

def gcd(a, b):
    if a == 0:
        return b
    return gcd(b%a, a)

for k in range(10000, 100000):
    c = 0
    c_last = 2
    for i in range(2, k+1):
        c = fast_power(c_last, i)
        c = c % N
        c_last = c
    c = c - 1

    p = gcd(c, N)
    # print "%s %s" % (p, c)
    q = N / p
    if p * q == N:
        print p

```

answer :

p = 582370792792333441832697622250283475151337365176 33 q = 148019364692112624967134248887606564835436496159 37