#### Part A

Check that the system parameters p, q, and g satisfy the three criteria in Table 1. You need to provide the factorization of p-1, and explain your verification process. Explain why we only pick ski as a number less than 224 bits. Compute pki, i=1,2,3.

A1 - p, a prime number lying between 1024 and 2048

```
# binary literal starts from index 2 onwards of bin(n) output

p = 1615850420240242625399113195036680055148205339919365512280505165762970604025264132936922942592721900695647374247690

len(bin(p)[2:])
2048

is_prime(p)
True
```

Figure 1.1 – Verification of A1: p is a prime number lying between 1024 and 2048 bits

#### A2 - q: a 224-bit prime factor of p - 1

```
q = 13479974306915323548855049186344013292925286365246579443817723220231

len(bin(q)[2:])
224

(p - 1) % q == 0

True

is_prime(q)

True
```

Figure 1.2 – Verification of A2: q is a 224-bit prime factor of p-1

## A3 - g: an element $g \in GF(p)$ with order q

g must satisfy the following criteria:

$$I. g^{p-1} mod(p) = 1$$

We can prove this criterion using Fermat's Little Theorem which states that:

For any modulus p and integer g coprime to p, one has

$$g^{\varphi(p)} \equiv 1$$

Where  $\varphi(p)$  denotes Euler's totient function (which counts the integers from 1 to p that are coprime to p). Fermat's little theorem is a special case, because if p is a prime number, then  $\varphi(p) = p - 1$ . Using Fermat's Little Theorem we get:

$$g^{p-1} mod(p) = 1$$

# $II. g^r mod(p) \neq 1 \forall 1 \leq r < p-1$

For this criterion, we only need to verify this case holds for all r that are prime factors of p-1 (assuming there are 3 prime factors):

Using SageMath's elliptic curve factorization module (ECM), we are able to perform a factorization of p-1 using the find factor method:

```
ecm.find_factor(p-1)

[2,

80792521012012131269955659751834002757410266995968275614025258288148530201263206646846147129636095034782368712384519
893943641863398312838663387479637823929232659368983403503877358201165266339339704498811349277692481146363231336300996659
753772809134790576619820837861560563416171843727915949442496818464040976140278193081677711641206211580015942455384765
144882595321111739393623341618105978256416706894225642006316565354661148064717778465381920470479616049441594917191871
183780972949256698364365971631232769775454026991957750963849972192971215891213833094419016280605610735416499107060455
47583106995719979463007803486771]
```

Figure 1.3 – Factorization of p-1 using the find factor method from SageMath's ECM module

We know p is an odd number, so p-1 is an even number and therefore its first prime factor must be 2. Then we check to see whether or not the other factor, u, returned by this function is a prime number:

```
u = 8079252101201213126995565975183400275741026699596827561402525828814853020126320664684614712963609503478236871238451
is_prime(u)
False
```

Figure 1.4 – Negative primality test of the factor u returned from the find\_factor method using p - 1

Since u is not a prime factor, we recursively divide p-1 by q (a prime factor of p-1) and verify if the second value returned by the find\_factor function is a prime number (since the first value returned is always 2). Fortunately, only one iteration is required before determining the other prime factor of p-1, v:

Figure 1.5 – Positive primality test of the factor v returned from the find\_factor method using (p-1)/q

We now have 3 prime factors for p-1: 2, q, and v and can verify the following criterion:

$$II. g^r mod(p) \neq 1$$

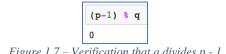
```
pow(g, 2, p) == 1|
False

pow(g, q, p) == 1
True

pow(g, v, p) == 1
False
```

Figure 1.6 – Verification of second criterion to determine if g is an element of GF(p)

Note that  $g^r mod(p) = 1$  for r = q. Lagrange's theorem states that the order of any subgroup of a finite group divides the order of the entire group. If g is any number coprime to p then g is in one of these residue classes. Thus, group element g has finite order q, and its powers g, g, ..., gk (mod p) form a subgroup of the group of residue classes, with gk = 1 (mod p). Consequently, Lagrange's theorem states that q must divide  $\varphi(p)$ .



Since q divides p-1, and because there exists no prime factor of p-1 lesser than q which satisfies this condition, we know that q is the order of the element g, which has also been verified as a primitive element of GF(p) given that the other prime factors of p-1 satisfied condition II.

### Why we pick sk<sub>i</sub> as a number less than 224 bits

ski is used in two places – computing the corresponding pki, and in the signing function. In each location, ski is used in an arithmetic operation which incorporates a modulo, meaning that regardless of the key value, the result of the operation is bounded from zero to the divisor minus one. As a result, extending the length of ski beyond the requirement does not provide any additional security.

pk<sub>i</sub> computation

For i = 1, 2, 3:

```
pk_1 = pow(g, sk_1, p)
pk_2 = pow(g, sk_2, p)
pk_3 = pow(g, sk_3, p)
```

*Figure 1.8 – Function used to compute pki* 

The above values can be found in the Appendix.