A **hash function** is a function that on input of any size, outputs a value of a fixed size. One use of the hash function is on data organization. How do you organize data so that they can be searched without going through all records?

(1) Consider the function $f(x) = x \bmod 7$.

- **(a)** Use Maple to generate a 100 random inputs between -1000 and 1000. Find their outputs and count the frequences of each output.
- **(b)** What can you conclude from the observations?

(2) Consider the similar function $f(x) = x \bmod 65537$. Without using any technology, can you find an x such that $f(x) = 1001$?

In general, a hash function should be easy to compute, such as the one mentioned above. However, for cryptographic purposes, we also need further restrictions. One such requirement is that we should not be able to find the *preimage* easily. That is, given an output y, it is difficult to find x where $f(x) = y$.

(3) Let $p = 7345363$. Consider the function $f(x) = x^{17} \bmod p$. This function can also be used as a hash function.

- **(a)** Find x such that $f(x) = 131072$.
- **(b)** Find x such that $f(x) = 5979767$.
- **(c)** Find x such that $f(x) = 5239027$.
- **(d)** What do you think about the security of this hash function?

(4) Evaluate $5239027^{3456641}$. Does this "magic power" work with other values of output y?

We can construct a similar has function as above using some RSA-alike features, that can be secure based on the difficulty of factoring a product of two primes. Based on your knowledge of RSA encryption algorithm, can you devise a scheme to turn it into a hash function?

(5) Use what you devised, make a hash function scheme, provide an output, and challenge your fellow group members to solve it.