Question 1.
Which of the following is not a property that you expect a well-designed hash function to have?
- The hash function should be easy to store (constant space or close to it).
- The hash function should "spread out" every data set (across the buckets/slots of the hash table).
- The hash function should "spread out" most (i.e., "non-pathological") data sets (across the buckets/slots of the hash table). - *This is something we'd like, but has not been achieved, so it is not expected.*
- The hash function should be easy to compute (constant time or close to it).

The hash function should "spread out" most (i.e., "non-pathological") data sets (across the buckets/slots of the hash table).

Question 2.
Suppose we use a hash function h to hash n distinct keys into an array T of length m. Assuming simple uniform hashing --- that is, with each key mapped independently and uniformly to a random bucket --- what is the expected number of keys that get mapped to the first bucket? More precisely, what is the expected cardinality of the set {k:h(k)=1}.
1/m per bucket since all are equally likely. The expected value is therefore, n * (1/m) = n/m
n/m

Question 3.
Suppose we use a hash function h to hash n distinct keys into an array T of length m. Say that two distinct keys x,y collide under h if h(x)=h(y). Assuming simple uniform hashing --- that is, with each key mapped independently and uniformly to a random bucket --- what is the probability that a given pair x,y of distinct keys collide?
Hashing keys are independent events, so the probability of a given pair of distinct keys hashing to the same bucket is 1/m * 1/m. Since hashing into each of the buckets is equally likely, the probability of a given **pair** of **distinct** keys hashing to **any** bucket is m * 1/m * 1/m = 1/m.
1/m

Question 4.
Suppose we use a hash function h to hash n distinct keys into an array T of length m. Assuming simple uniform hashing --- that is, with each key mapped independently and uniformly to a random bucket --- what is the expected number of pairs of distinct keys that collide? (As above, distinct keys x,y are said to collide if h(x)=h(y).)
There are n choose 2 pairs of distinct keys and each pair has a 1/m chance of colliding. So that means the expected number of pairs of distinct keys that collide is (n * (n - 1)) / 2*m.
(n * (n - 1)) / 2*m

Question 5.
To interpret our heuristic analysis of bloom filters in lecture, we considered the case where we were willing to use 8 bits of space per object in the bloom filter. Suppose we were willing to use twice as much space (16 bits per object). What can you say about the corresponding false

positive rate, according to our heuristic analysis (assuming that the number kk of hash tables is set optimally)? [Choose the strongest true statement.]

Less than .1%