You will submit your solution to this assignment to the Curator System (as `HW02`). Your solution must be either a plain text file (e.g., NotePad++) or a typed MS Word document; submissions in other formats will not be graded.

Partial credit will only be given if you show relevant work.

---

1. Suppose you are implementing a hash table and are trying to choose between using a probing strategy and using chaining (each slot is actually a linked list that can hold as many records as needed, prepended to the list).

You will use the same hash function and hash table size no matter which strategy you select.

a) **[10 points]** A *primary collision* occurs when two records map to the same home slot in the table. Assuming you insert the same set of records into the hash table, in the same order, how will the number of primary collisions be affected if you switch from quadratic probing to chaining? Justify your conclusion carefully.

b) **[10 points]** One objection to using chaining is that you'll have to perform a linear traversal of the linked list to find a record, and linear traversals are slow. Considering that, and assuming that a small but significant number of primary collisions will occur, would searching be more efficient if you chose quadratic probing instead of chaining? Justify your conclusion carefully.

2. A hash table implementation uses quadratic probing to resolve collisions. Suppose the table was initially empty, and then records S1, S2, …, and SM were inserted into the table, in that order, all of those records have different key values, but all of those records map to the same home slot.

Then, N additional records R1, R2, …, and RN were inserted into the table, in that order, all of those records have different key values, and all of those records map to the different home slots from each other, none of which are the home slot of the Sk that were inserted earlier.

a) **[10 points]** What is the smallest number of record comparisons that could be performed in a search for the record RN. Justify your conclusion.

b) **[10 points]** What is the largest number of record comparisons that could be performed in a search for the record RN. Justify your conclusion.

3. Assume the same situation as in question 3, except that chaining is used to resolve collisions, with linked lists for the chains.

a) **[10 points]** What is the smallest number of record comparisons that could be performed in a search for the record RN. Justify your conclusion.

b) **[10 points]** What is the largest number of record comparisons that could be performed in a search for the record RN. Justify your conclusion.

4. Consider a hash table consisting of $M = 13$ slots, and suppose nonnegative integer key values are hashed into the table using the hash function `h2()`:

```
uint32_t h2(uint32_t key) {

   uint32_t prime = 0x811C9DC5;
   uint32_t hashvalue = 0;
   uint32_t digit;

   while ( key > 0) {
      digit = key % 10;
      hashvalue *= prime;
      hashvalue ^= ( digit + '0');
      key = key / 10;
   }
   return hashvalue;
}
```

You are strongly advised to write C code to use the function above to hash the key values given below.

a) [20 points] Suppose that collisions are resolved by using linear probing. The integer key values listed below are to be inserted, in the order given. For each key, show the home slot (the slot to which the key hashes, before any probing), the sequence of slots that are examined (including the home slot and any slots examined if probing is necessary), and the final contents of the hash table after the given key values have been inserted in the table:

Key Value	Home Slot	Slots examined during insertion
146		
152		
215		
303		
418		
116		
149		
352		
232		
182		

Total number of slots examined: _______

Final Hash Table:

Slot	0	1	2	3	4	5	6	7	8	9	10	11	12
Contents													

b) **[20 points]** Repeat the previous question, but now resolve collisions using quadratic probing, with the probe function:

$$\frac{k^2 + k}{2}$$

Key Value	Home Slot	Slots examined during insertion
146		
152		
215		
303		
418		
116		
149		
352		
232		
182		

Total number of slots examined: _______

Final Hash Table:

Slot	0	1	2	3	4	5	6	7	8	9	10	11	12
Contents													