ECE 250 *Algorithms and Data Structure*
Department of Electrical and Computer Engineering
University of Waterloo
Please send any comments or criticisms to dwharder@alumni.uwaterloo.ca
with the subject ECE 250 Questions 9.9.
Assistances and comments will be acknowledged.

9.9*a* Describe the difference between double hashing and linear probing.

9.9*b* Why must the second hash value be relatively prime to the first hash value?

9.9*c* If we require that the size of the hash table is a prime number p, the first hash function will return a value in the range 0, …, $p - 1$. What are the restrictions on the second hash value?

9.9*d* If we require that the size of the hash table is a power of two, say 2^m, the first hash function will return a value in the range 0, …, $2^m - 1$. What are the restrictions on the second hash value?

9.9*e* In linear probing, when erasing an entry, we could step forward and rearrange the remaining entries so that we need only concern ourselves with occupied an unoccupied entries. With double hashing, we must mark erased entries. What is the justification for this?

9.9*f* When inserting a new entry into hash table using double hashing, is it reasonable or unreasonable to insert the new entry into a bin marked as erased?

9.9*g* Suppose a hash function generates the 32-bit values 2509325 and 7103923 for two objects. Multiply each of these values by the prime number 3221225473 and calculate the appropriate modulo to find the bin in a hash table of size 2^{10}. Then, multiply the original hash values by the prime number 1073741827 and calculate appropriate jump sizes.

9.9*h* Insert the following 16 hexadecimal numbers into a hash table of size 16 where the least significant hexadecimal digit (a *nibble*) is used to determine the bin, and the second least significant nibble is used to determine the jump size.

EF6, 7DF, FE1, 92B, 625, D1A, 810, 6E4, AFA, 511, 137, 652, A59, F3B, D8D, 97A

0	1	2	3	4	5	6	7	8	9	10/A	11/B	12/C	13/D	14/E	15/F

9.9*i* Given the hash table generated in Question 9.9*h*, what is the average number of bins that must be searched to find the 16 entries?

9.9*i* From the hash table built in Question 9.9*h*, erase the entries EF6, 7DF, FE1, 92B, 625 and D1A.

9.9*j* What is the load factor of the hash table in Question 9.9*h*. Why should this never be done in practice?

9.9*k* Inserting the first 12 entries results in the hash table

0	1	2	3	4	5	6	7	8	9	10/A	11/B	12/C	13/D	14/E	15/F
810	FE1	511		6E4	625	EF6	137		AFA	D1A	92B	652			7DF

What is the average number of searches required to find each of the twelve entries?

9.9*l* Insert the following 12 hexadecimal numbers into a hash table of size 8 where the least significant hexadecimal digit (a *nibble*) is used to determine the bin, and the second least significant nibble is used to determine the jump size. If an insertion causes the load factor to increase above $\rho = 0.75$, rehash the values currently stored in the hash table (in the order in which they are stored in the bins) into a new hash table of double the size, and then continue inserting the next entry. Show your work by showing the intermediate hash table of size eight before doubling the size of the hash table.

807, 4AD, 988, CBA, 380, C4C, 426, A46, ECD, 032, D8B, DBE

0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F

0	1	2	3	4	5	6	7	8	9	10/A	11/B	12/C	13/D	14/E	15/F

9.9*m* From the hash table generated in Question 9.9*l*, erase the entries 807, 4AD, 988, CBA, 380, C4C, 426, A46, ECD and if, after an erase, the load factor drops below $\rho = 0.3$, half the size of the hash table and enter the values currently stored in the hash table (in the order in which they are stored in the bins) into a hash table of half the size, and then continue erasing the entries. Show your work by showing the intermediate hash table of size eight before doubling the size of the hash table.

0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F

9.9*n* Explain why we would not want to double the hash table if the load factor goes above $\rho = 0.75$ and then half the capacity if the load factor goes below $\rho = 0.375$.

9.9*o* According to the formulas present in class, what is the expected number of probes for both a successful and an unsuccessful search if do not allow the load factor is not allowed to exceed $\rho = 0.\ 75$? What is the expected number of such probes if we do not allow the load factor to exceed $\rho = 0.8$.