19. Pseudo keys are in the range 0 to 200,000 and a maximum of 9000 nodes will be stored in a non-perfect hashed data structure. Give the size of the primary storage area array (number of elements) that will produce the optimum loading factor.

For a non-perfect structure the optimum loading factor is 3. So the optimum size is 9000/3 = 3000.

20. Give the density of the structure described in Exercise 19 assuming the node width is:

a) 10 bytes

D = 1/(1+8/w+(4/(3w))) = 0.517.

b) 200 bytes

D = 1/(1+8/w+(4/(3w))) = 0.955

21. Assuming keys are comprised of upper- and lowercase letters, give the bit pattern and base 10 numeric value of the pseudo key that would result from processing the key “Mary” using the algorithm illustrated in Figure 5.17.

M a r y

x4 D 6 1 7 2 7 9 (in Hexadecimal)

0100,1101,0110,0001 ,0111 ,0010,0111,1001 (Binary)

1,298,231,929 (Decimal).

22. What would be the base 10 value of the pseudo key produced by fold-shifting the key “Bob-Jones” to produce a 16-bit numeric pseudo key. Use the characters “bJ” as a pivot.

Wondering about interpretation on this one. We are starting with a string of 9 characters. That is 72 bits when written as translated unicode. It would follow the algorithm closely to go for a 24 bit pseudo-code using "-Jo" as a pivot. An easy way to get 16 bits would be to then truncate the high-order bits. The result is: 24 bits: xDE1F44 = 1,456,996 = 1101,1110,0001,1111,0100,0100 truncated: x1F44 = 8,004 = 0001,1111,0100,0100.

A possible second method in the same spirit as the fold-shift breaks that 72 bit number up as 12/16/16/16/12 bits. These five numbers can then be added. (The breaks are all in the middles of letters) The result is: xCB47 = 52,039 = 1100,1011,0100,0111 (A carry was ignored in the last step).

A third option would be to use ASCII instead of unicode. This gives 63 bits from the string (only 7 bits per character. The resulting bit pattern could be broken up as 15/16/16/16, etc. This loses all connection between the boundaries used in processing and the characters. (ASCII characters aren't strung together in 7-bit bytes either. It's nice theoretically but isn't a practical solution)

23. A pseudorandom preprocessing scheme is being used in a hashed data structure with the primes p1 = 11 and p2 = 5. Give the pseudo keys for the following keys:

a) 198 (198\*11+5) = 2183

b) 24 (24\*11+5) = 269

24. State the difference between an open addressing and a non-open addressing collision algorithm.

Open addressing generates a new address in the main storage area. Non-open addressing creates secondary storage area outside of the main storage area. (This is often a LinkedList but is not required to be.)

25. The primary storage area in a non-perfect hashed data structure is a 103 element array. A key has been mapped into index 102, and a collision has occurred. Give the indices calculated by the next three passes through the collision algorithm if the collision algorithm is the modified version of the:

a) Linear Probing

0,1,2

b) Quadratic Probing

0,4,13

26. Define the terms:

a) Primary clustering

Primary clustering occurs when one is using a linear collision algorithm. It fills up a region of the main array. Any attempt to add a node in the midst of this region will cause more collisions.

b) Secondary clustering

Secondary clustering happens with the quadratic algorithm. There is still a sequence of nodes that must be followed when a primary collision occurs. Collisions in the chain from other primary keys won't follow the same path though.

27. Why is the array used for a restricted hashing scheme always sized to a 4k + 3 prime?

Prime numbers are preferred because the process of remaindering wrt a prime will result in an even distribution of pseudo-keys. I can't find any reliable information on why it should be a 4k+3 prime.

Primes of the form 4k+1 can always be written as the sum of two squares. Primes of the form 4k+3 can never be.

Here's a point that may actually matter. Primes of the form p= 4k+3 cause the mapping x^2 to permute the quadratic residues mod p. Primes of the form 4k+1 don't do this. For instance: If p=7, the quadratic residues are 0,1,2,4. The quadratic non-residues are 3,5,6. Look at x^2: 0->0,1->1, 2->4, 4->2, permutes the quadratic residues.

when p=5, the quadratic residues are 0,1,and 4. 0->0, 1->1 and 4->1 under x^2. Which is a valid function but is not a permutation. Does not permute the quadratic residues.

In both cases the quadratic non-residues are not permuted by x^2.

28. Give the Linear Quotient collision algorithm and describe how it gets its name?

before execution we have ip = pk%p = r <-> pk = p\*q+r // p is a prime.

if(q%p == 0)

ip = (ip+q)%p;

else

ip = (ip+p')%p;

Collisions occur when there is a duplicate pseudo-key. By the time the collision method is called there is already a collision and we have a duplicate key. That key is a remainder from a previous division by a prime p. Call the quotient from that calculation q. There are then two choices. Either p|q or it doesn't. If p|q then take another prime p' and use that as an offset and remainder with p to find a new key. Otherwise offset by r and remainder with p to get the new key.

29. Describe the “delete problem” associated with non-perfect hashed data structures and state the standard remedy.

Deleting a node that has other nodes "behind it" makes those other nodes inaccessible. To solve this a blank link or a link to a dummy node is inserted instead of the now missing node.

30. Give the density of a hashed structure with a loading factor of 0.6 assuming each node contains 30 information bytes.

We have L = 0.6 and w=30. D = w/(w+4/L) = 0.81

31. Objects to be stored in a data structure each have 1000 bytes of information, which includes the key field comprised of 5 digits. A maximum of 1500 nodes will be in the structure at one time. How many elements will be in the primary storage area array, if the nodes are stored using a:

a) Direct hashing function?

There must be 10^5 = 100,000 elements for direct hashing.

b) Division hashing function?

For division hashing we need a prime number larger than 1500. The proper choice is 1511 as the smallest 4k+3 prime larger than 1500.

Give the density of the two structures described in Exercise 31.

For direct hashing the loading factor is 1500/10^5 =0.015. This gives a density of

D = w/(w+4/L) = 0.79.

For the division hashing function the loading factor is 1500/1511 = 0.992

The density is D = 0.996.

Not really a surprise that division hashing has a better density.

33. A maximum of 300 nodes are to be stored in a hashed data structure. Give the size of the primary storage area that would maximize the performance of the structure.

lmax = 0.75 -> N > (300\*1.75) = 525. N = 547 is the smallest 4k+3 prime larger than 525.

34. A 23 element array has been allocated to store nodes using the LQHashed data structure discussed in this chapter. Give the array index used to store the nodes with the following keys, assuming they are inserted in the order given:

a) 4618 index = 18.

b) 391 index = 0.

c) 6941 q=301, r=18 -> q=13, r = 2 index = 2; (p=23) (301=13\*23+2).

d) 547 q=23 r=18 (547 = 23\*23 +18) -> (18+19)%23 = 14 index = 14.

Note: Use the 4k + 3 prime 19 to resolve any problems with the quotients.

35. Give the advantage of dynamic hashed structures over non-dynamic hashed structures.

The main advantage of dynamically hashed structures is the fact that memory can be expanded.

36. Give an advantage and a disadvantage of the array-based approach to a dynamic hashed structure over the linked approach.

The main advantage of the array based approach has to do with the ease of encapsulation.

The main disadvantage is the difficulty of expanding the size of the structure.

37. In the context of dynamic hashed structures, what is a bucket?

At each address of the main array is a structure (usually a linked-list) that contains all the nodes for that address. This structure is called a bucket.

38. Assuming 30,000 nodes are equally distributed over a dynamic linked hashed structure, give the size of the primary storage array for optimum performance.

The book says that the optimum pseudo-loading factor is 3. So N = 30000/3 = 10000. It would probably be better to choose the prime number 10039 which is the smallest 4k+3 prime greater than 10000.

39. Java contains a class that implements a dynamic hashed data structure.

a) Give the name of the class.

The book discusses the Hashtable class. The HashMap also could be used for this. It depends on what one is doing which of these will be better. There is also HashSet. All three are dynamically hashed structures.

b) Give the import statement necessary to use the class.

There are several choices that will work. I use import java.util.\*; in this course, we are using many classes from java.util.

c) How does it resolve collisions?

Mashtable, HashMap and HashSet all create a linked list at each address. Collisions move down the list.

d) Is the structure a fully encapsulated structure?

Generally, no.