

Name: (as it would appear on official course roster)		
Umail address:	@umail.ucsb.edu	section 9am, 10am, 11am, 12pm
Optional: name you wish to be called if different from name above.		
Optional: name of "homework buddy" (leaving this blank signifies "I worked alone")		

h09: Chained Hashing

ready?	assigned	due	points
true	Tue 10/29 02:00PM	Tue 11/05 02:00PM	50

You may collaborate on this homework with AT MOST one person, an optional "homework buddy".

MAY ONLY BE TURNED IN IN THE LECTURE LISTED ABOVE AS THE DUE DATE,
OR IF APPLICABLE, SUBMITTED ON GRADESCOPE. There is NO MAKEUP for missed assignments;
in place of that, we drop the lowest scores (if you have zeros, those are the lowest scores.)

Reading: Chained Hashing, DS 12.3

- (10 pts) Fill in the information in the header. The following are required to get the 10 "participation" points.
 - Filling in your name and umail address.

Also: For paper submission PLEASE submit on ONE SHEET OF PAPER, double-sided if at all possible. If you must submit on two printed sheets write name on BOTH sheets and no staples, paperclips, or folded corners.

Please also read the handout <http://cs.ucsb.edu/~richert/cs32/misc/h09-handout.pdf>

- Hash Functions are used when the main objective is to make searches as fast as possible, i.e. an expected time $O(1)$ search operation. Consider the typical process of finding an element: an $O(1)$ hash function $H(x)$ computed on an item x , giving some numerical location for a program to search for a value. A hash collision in a chained hash table creates a linked list of all elements that collide at any one index i .
 - (5 pts) The phrase **load factor** is a technical term that arises in this context. Briefly: what does this term refer to?

- (5 pts) If something goes wrong, a chained hash strategy easily lead from an average case $O(1)$ search operation to average case $O(n)$ search operation. What is the circumstance in which things could go awry in this way? **Briefly** describe.

1
h09
CS32 F19

3. Suppose that you are going to implement (without use of the STL) a chained hashing scheme for objects of class `Student`, using a hash over the perm number, using chained hashing. Suppose that a constant `ARRAY_SIZE` has been defined as the size of an array, and each element in that array should be the head of a linked list of `Students` that hashes to the index of that element.

a. (5 pts) Write a C++ definition for a struct that can represent a node in each of these linked lists.

b. (5 pts) Write a line of C++ that, if it appeared in the `private:` section of the class definition for `class StudentHashTable`, would represent a suitable array for the values, assuming that array was allocated on the heap in the constructor of `class StudentHashTable`

c. (10 pts) Write the definition of the constructor for `StudentHashTable` as it would appear in the `.cpp` file. Be sure to both allocate the space for the array on the heap, and set all of the pointers in that array to null.

4. (10 pts) Given the series of input [45, 7, 46, 90, 34, 6, 7, 71, 23, 24], insert each of these integers into the chained hash table below, with a hash function defined by $H(x) = x \% \text{tableSize}$. Draw in the linked lists to the right of each array index, showing a pointer to null as the end of each list. The Handout for this assignment has examples of the format we want for your answer.

0	
1	
2	
3	
4	

2

h09
CS32 F19