

HW1 – DUE 10 AM 14.JAN.2013

Instructions:

Place all of the definitions into a single file named HW1_[lastname].hs and drop it into your Assignments folder on Google Drive.

Note the elements specified below must be comparable.

1. Using a list comprehension, write a Haskell function “delete_all” that takes two parameters: an element x and a list y; delete_all returns the list y with all elements x deleted.
delete_all 1 [1,2,1] = [2]
2. Using a list comprehension, write a Haskell function “delete_second” that takes two parameters: an element x and a list y; delete_second returns the list y with the second occurrence of element x deleted. delete_second 1 [1,2,3,1,2] = [1,2,3,2]
3. Using a list comprehension, write a Haskell function “associated” that takes two parameters: an element x and a list of pairs, y; associated returns a list of the second elements of each pair in y that has a first element x. associated 3 [(3,4),(5,7),(3,6)]=[4,6]
4. Write a Haskell function sumUp that takes two parameters (both integers) and returns the sum of all the numbers between the two parameters (inclusive). You cannot make any assumptions about the order of the parameters. For example, sumUp 5 3 returns 12.
5. Write a Haskell function count that takes two parameters: an element x and a list y; count returns the number of times x occurs in y. count 3 [1, 3, 2, 3] = 2.
6. Write a Haskell function declist that takes one parameter: a list of numbers y; declist returns a list of numbers with each number one less than the corresponding number in the original list y. For example, declist [1, 2, 3, 4] = [0, 1, 2, 3].