## **CSC 262** Homework #11 Due at start of class on Wednesday, May 2.

1. Consider a program that must manipulate the elements of a very large twodimensional array. (Assume that "very large" means that the entire array will not fit at once in main memory.) Apply what we have learned about locality and working sets to answer the following questions.

a. Suppose that multiple operations must be performed on each array element. Programs A and B both make the same computations, but they differ in the order in which they do so. Program A completes all the operations on the first element before moving to the second element. Program B completes the first operation on all array elements before moving on to the second operation. Do you expect any difference in execution speed between the two programs? Why or why not?

b. A 2-d array will typically be stored in memory in one of two orders: row-major or column major. (Row-major means that the rows are stored together one after another; column-major means that the columns are stored together one after another.) Processing will also take place in either row-major or column-major order. Do you expect any difference in speed when the storage order matches the processing order, compared to when different orders are used? Why or why not? Which would require a larger working set?

1	2	3	4	
5	6	7	8	
9	10	11	12	
Row-major order				

1	4	7	10
2	5	8	11
3	6	9	12
<b>a</b> 1			

Row-major order

Column-major order

2. Suppose that a hypothetical system uses the buddy system (non-lazy version) for memory allocation. Assume that the system has 16K units in all, and the base allocation unit is 1K. Further assume that all searches for memory slabs start from the top of memory, using the first available block of the appropriate size (if any) before looking for a larger block to split. For the following sequence of memory requests, draw the state of memory at each (\*). Be sure to show boundaries between slabs of different sizes.

J1: 3K; J2: 5K; J3: 0.5K; (\*) J4: 1K; J5: 1K; J6: 1K; (\*) J4, J5, J2 finish; J7: 1.8K; (\*) J6 finish; J8: 2K; (\*)