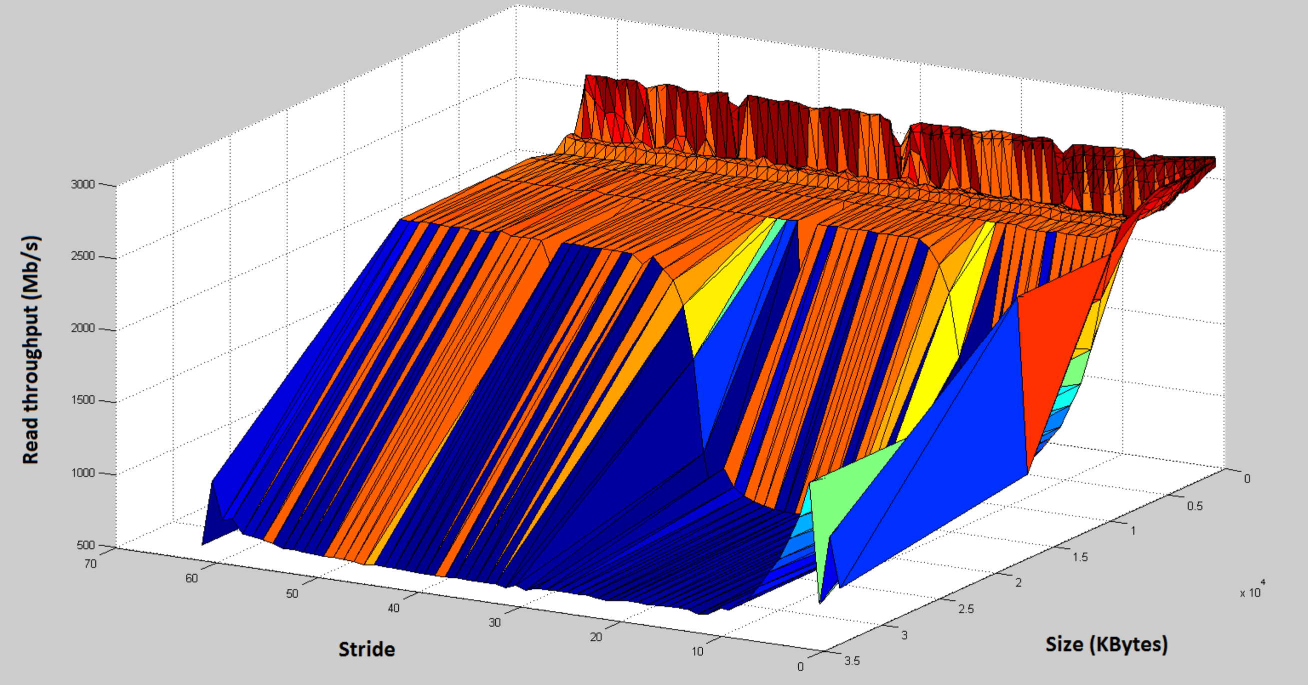
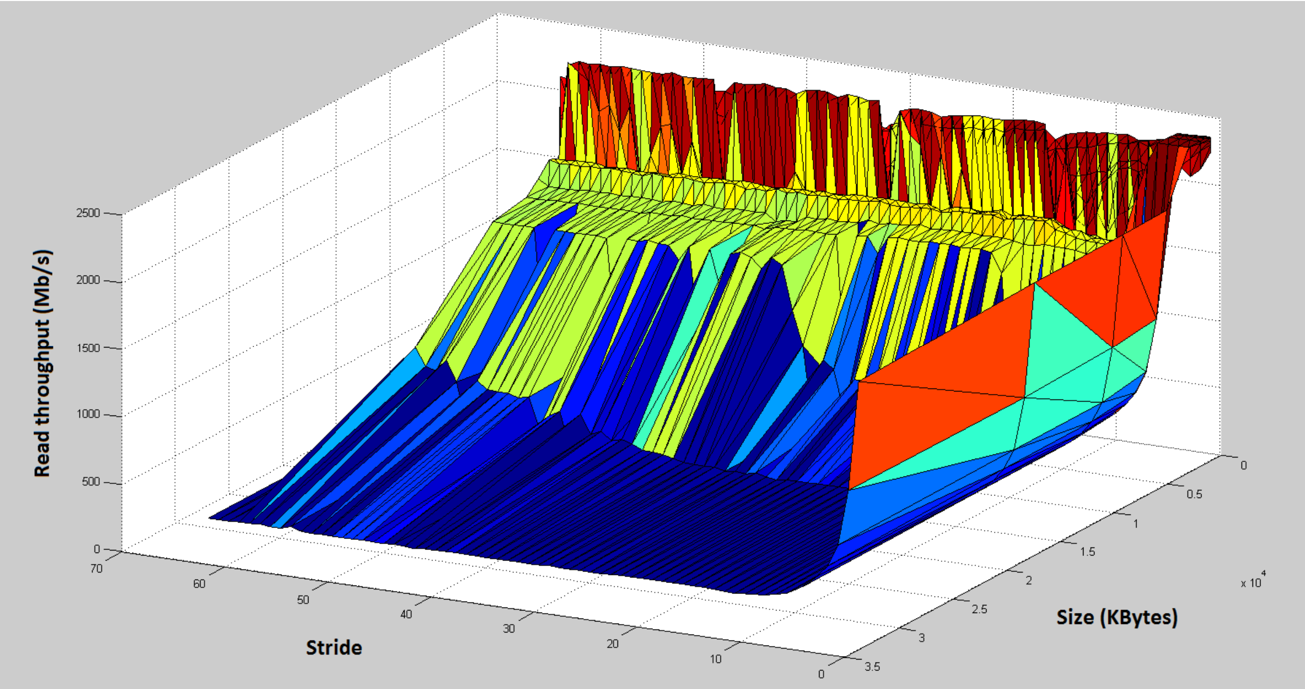
CPSC 313 – Assignment 5
Names: Alvin Shen 74164096, x3o7 and
Billy Lau 34102103, z4o7

2.
Linux : deas.ugrad.cs.ubc.ca
Memory mountain



If the stride values are increasing, then that means that spatial locality is decreasing. This can be seen by the downwards slope from strides 30 to 64. Whereas for decreasing stride values, spatial locality increases (i.e. from 0 to 30). We notice three ridges: the flat sections colored in blue, orange, and red (at the very top of the mountain). The top ridge would be the L1 cache, the orange ridge would be the L2 cache, and the blue is most likely the memory.

Mac Memory Mountain



The difference with this graph is that spatial locality decreases at a slower rate for large stride values. Unlike the previous graph, the slope is not as steep from strides 30 to 64. We see 4 ridges: the flat sections colored blue at the bottom, blue in the middle, yellow, and red at the top. These would be the memory, L3, L2, and L1 caches respectively.

3a) The miss rate is 0.1250. When a[0][0] is cold missed, a[0][0] to a[0][7] gets loaded into the line of set1. We can then see that a[0][1], a[0][2], a[0][3], …, a[0][7] will all be cache hits. This type of pattern repeats itself for subsequent values of i and j. Thus, there is a 1/8 = 0.1250 miss rate.

3b) The miss rate is 0.5000. When a[0][0] is cold missed, a[0][0] to a[0][7] gets loaded into the line of set1. a[0][1] gets a cache hit. When a[1][0] is conflict missed, a[1][0] to a[1][7] gets loaded into the line of set1. a[1][1] gets a cache hit. This type of pattern repeats itself for subsequent values of i and j. Thus, there is a 2/4 = 0.5000 miss rate.

3c) The miss rate is 1.0000. When a[0][0] is cold missed, a[0][0] to a[0][7] gets loaded into the first line of set1. Afterwards a[1][0] is cold missed as well, and a[1][0] to a[1][7] gets loaded into the line of setX. Then a[2][0] is gets conflict missed, and a[2][0] to a[2][7] gets loaded into the line of set1. Similarly for a[3][0]. This type of pattern repeats itself for subsequent values of i and j. Thus, there is a miss rate of 1.0000.

3d) The miss rate is 1.0000. When a[0][0] is cold missed, a[0][0] to a[0][7] gets loaded into the first line of set1. Afterwards a[1][0] is cold missed as well, and a[1][0] to a[1][7] gets loaded into the second line of set1. Then, a[2][0] gets a conflict miss, and a[2][0] to a[2][7] gets loaded into the first line of set1. Similarly, a[3][0] also gets a conflict miss, and a[3][0] to a[3][7] gets loaded into the second line of set1. We can then see that for j >=1 there will be all conflict misses, thus the 1.0000 miss rate.

3e) The miss rate is 0.1250. When a[0][0] is cold missed, a[0][0] to a[0][7] gets loaded into the first line of set1. Afterwards a[1][0] is cold missed as well, and a[1][0] to a[1][7] gets loaded into the second line of set1. Then a[2][0] is cold missed, and a[2][0] to a[2][7] gets loaded into the third line of set1. Similarly, a[3][0] also gets a cold miss, and a[3][0] to a[3][7] gets loaded into the fourth line of set1. Now for 0 < j < 8, everything will be cache hits. In other words, for every 8th iteration of j, there will be cache misses. Thus 1/8 = 0.1250 is the miss rate.

3f) The miss rate is 0.1250. When a[0][0] is cold missed, a[0][0] to a[0][7] gets loaded into the line of set1. Afterwards a[1][0] is cold missed as well, and a[1][0] to a[1][7] gets loaded into the line of setX. Then a[2][0] is cold missed, and a[2][0] to a[2][7] gets loaded into the line of setY. Similarly, a[3][0] also gets a cold miss, and a[3][0] to a[3][7] gets loaded into the line of setZ (this line does not overlap any of the existing lines). Now for 0 < j < 8, everything will be cache hits. This pattern repeats for j >= 8. Thus there is a miss rate of 1/8 = 0.1250.

Time spent: 20 hours