Solomon Heisey

Project 3 Writeup

Below is the raw data resulting on running the following splits on the following memory configurations. All results were ran on the 1.trace file. Note: memory accesses were discarded since it was a constant among all cases.

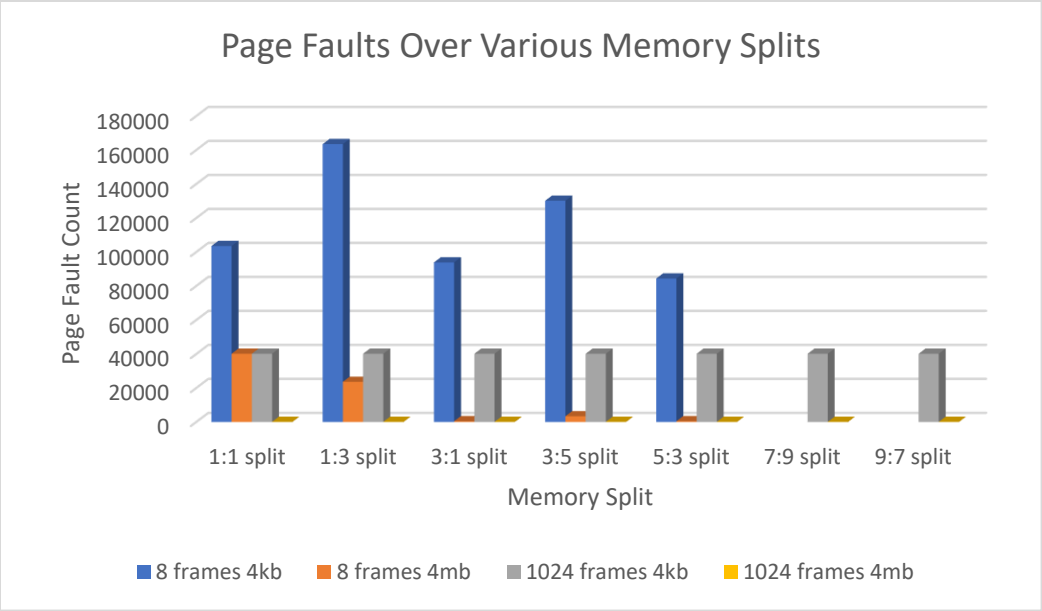
Page Faults Over Various Memory Splits

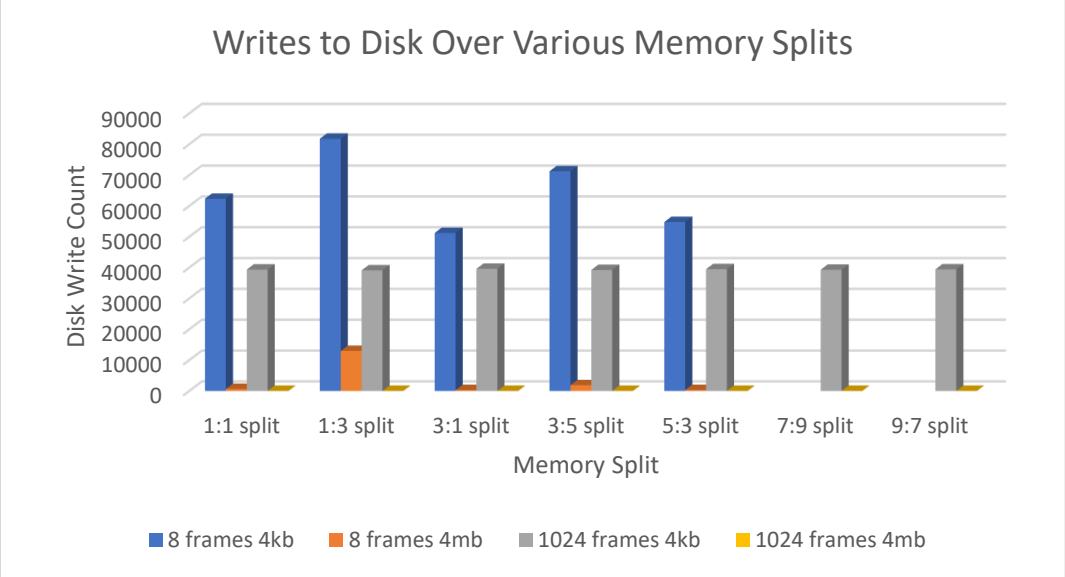
	1:1 split	1:3 split	3:1 split	3:5 split	5:3 split	7:9 split	9:7 split
8 frames 4kb	103539	163510	93875	130174	84408		
8 frames 4mb	40168	23635	452	3391	502		
1024 frames 4kb	40168	40169	40168	40168	40168	40168	40168
1024 frames 4mb	94	94	94	94	94	94	94

Writes to Disk Over Various Memory Splits

	1:1 split	1:3 split	3:1 split	3:5 split	5:3 split	7:9 split	9:7 split
8 frames 4kb	62306	81767	51215	71222	54757		
8 frames 4mb	639	12964	270	1861	306		
1024 frames 4kb	39342	39090	39598	39214	39470	39278	39406
1024 frames 4mb	0	0	0	0	0	0	0

Based on the above tables, the following bar charts were generated:





Analysis of data: What the data above shows is that when page size is constant, more frames led to an overall reduction in page faults as well as overall disk writes. When frame size is constant, more memory also leads to an overall reduction in the number of page faults and disk writes since each frame is larger, so it can store more data. Therefore, increasing page size and frame count necessarily leads to less page faults and less disk writes. I did not mention memory splits in my analysis since it is dependent on the input trace file, so some splits may perform better or worse depending on the input file.