1. Drew Sadler
2. Create A: | 5 | 15(free) |
   Create B: | 5 | 3 | 12(free) |
   Create C: | 5 | 3 | 3 | 9(free) |
   Create D: | 5 | 3 | 3 | 5 | 4(free)|
   Delete B: | 5 | 3(free) | 3 | 5 | 4(free)|
   Delete C: | 5 | 3(free) | 3(free) | 5 | 4(free)|
   Create E: | 5 | 4 | 2(free) | 5 | 4(free)|
   Delete A: | 5(free) | 4 | 2(free) | 5 | 4(free)|
   Create F | 3 | 2(free) | 4 | 2(free) | 5 | 4(free)|
3. There are 2 chunks of 2 or less in the memory
4. 4 blocks free/ 20 total blocks = 20% fragmentation
5. Create A: | 2 | 18(free)|
   Create B: | 2 | 4 | 14(free)|
   Create C: | 2 | 4 | 1 | 13(free)|
   Extend A: | 5 | 4 | 1 | 10(free)|
   Reduce B: | 5 | 2 | 1 | 12(free)|
   Create D: | 5 | 2 | 1 | 5 | 7(free)|
   Delete C: | 5 | 2 | 1(free) | 5 | 7(free)|
   Delete A: | 5(free) | 2 | 1(free) | 5 | 7(free)|
   Create E: | 4 | 2 | 2(free) | 5 | 7(free)|
6. There are 2 blocks that have size 2 or less
7. I'm not sure since this method also ends up with 20% fragmentation, but allows for a majority of free space to exist in the last block
8. 3 blocks must be read in order to find the byte of the file

File Allocation Table(FAT)				
5	2	1	5	7 (free)
A	B	C	D	E

5	2	1 (free)	5	7 (free)
A	B	C	D	E

5 (free)	2	1 (free)	5	7 (free)
Free drive	B	C	D	E

4	2	2 (free)	5	7 (free)
E	B	C	D	F/free drive

9.
10. 1KB block = 10,485,760 FAT entries
    4KB block = 2,621,440 FAT entries
    8KB block = 1,310,720 FAT entries
11. 1KB block = 31,457,280 bytes
    4KB block = 7,864,320 bytes
    8KB block = 3,932,160 bytes
12. 1KB block = 10 + 3*(3) = 19 microseconds
    2KB block = 10 + 3*(2) = 16 microseconds
    4KB block = 10 + 3*(1) = 13 microseconds