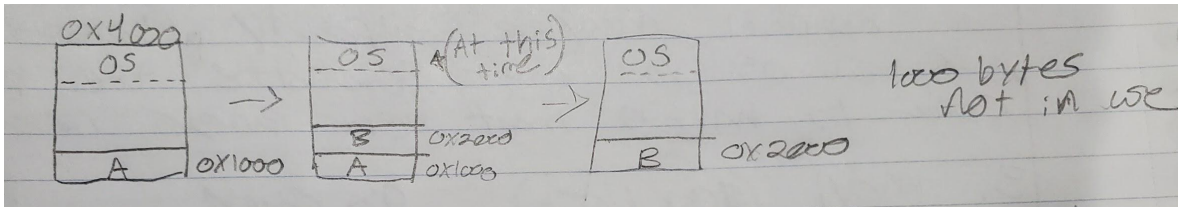
1. Drew Sadler
2. They would likely have a problem since programs are usually stored in boundaries, and since it would be within this range rather than at start, it could be inside of a program that may already be placed there. As well as you must relocate the movl code to within that new range manually.
3. Movl ($0x1100), %eax
   Movl ($0x1104), %ebx
   Movl %eax, ($0x1104)
   Movl %ebx, ($0x1100)
4. 0x0 -> 0x1023
   0x1024->0x2047
   0x2048->0x3071
   0x3072->0x4095
   Ending on the 4096 byte?
5. * Nothing would happen as it has been moved out of range
6. No, since it would produce an out of range segmentation fault, and programs cannot interfere with each other. It could relocate quite easily by adding the base register to the virtual address, so it could be able to end up in the right place.
7. The process of the movl moves content from the 2nd program to the 1st program, with the jmp function keeping it within range of the base and limit, so nothing would be able to happen outside of it.
8. 

9. As program B is still running and has to wait till finish or release. Or the programs were not contagious and is able to fragment/compact correctly when swapping so no space is left in memory
10. Partition memory to make space to stack and swap between the programs quicker as memory was already allocated to it, or by fragmenting the memory space and allowing for dynamic allocation to be able to make more space