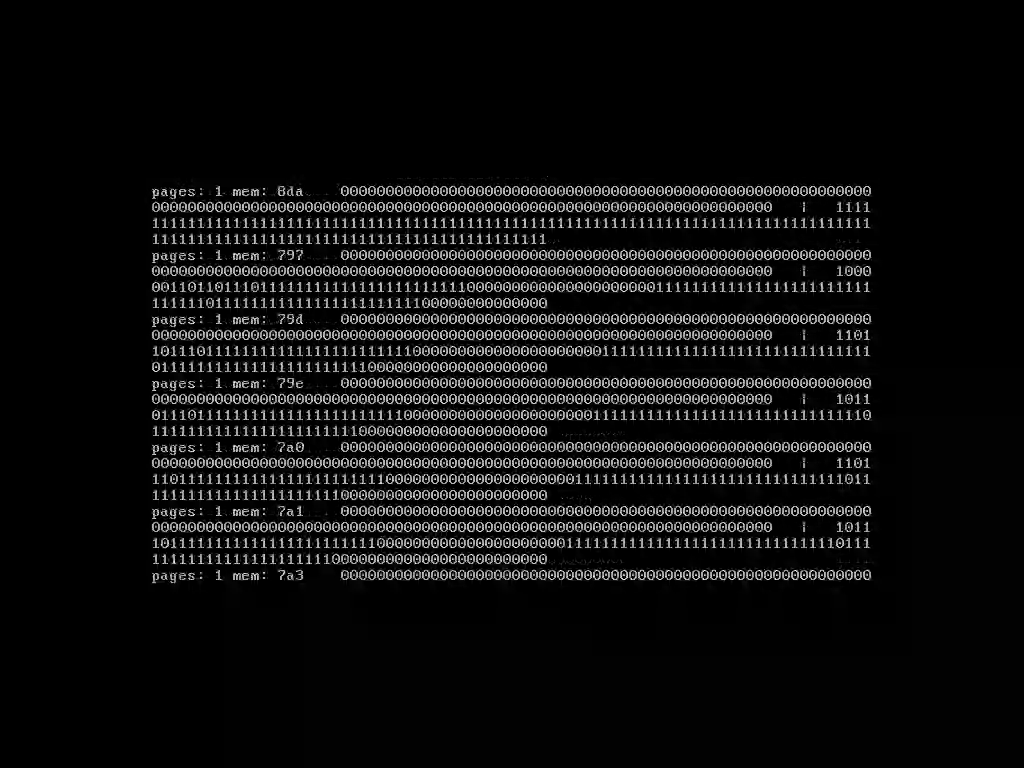
Josh Wretlind

Chris Sanford

12/02/13

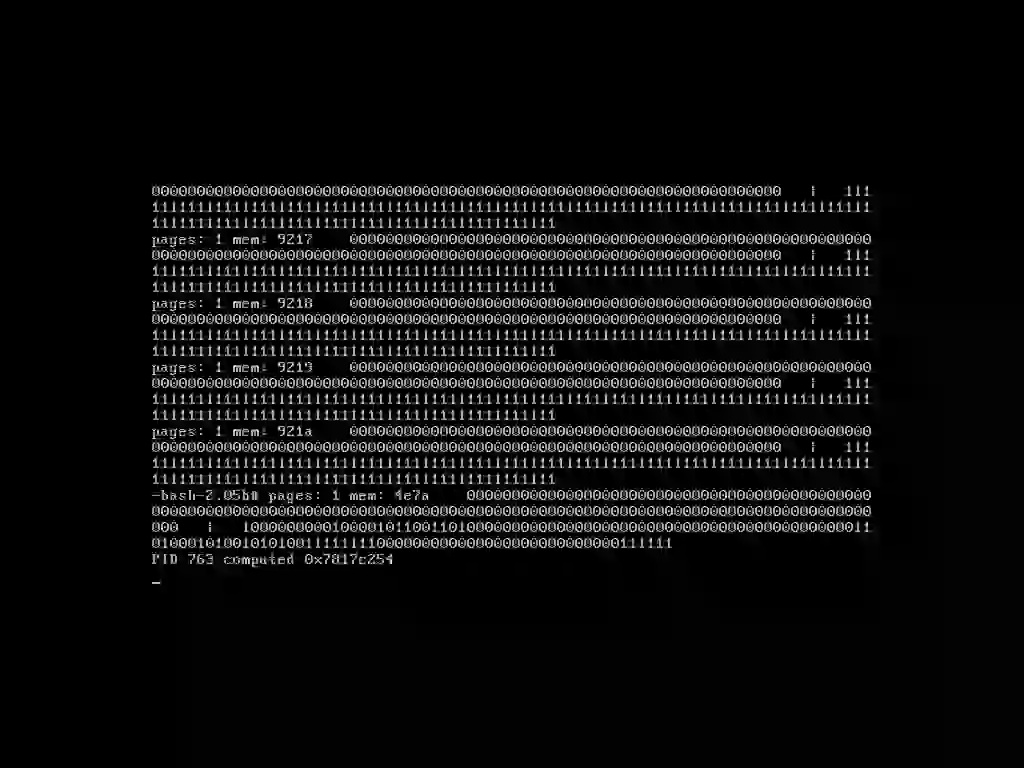
CSCI442

Minix originally had used Next fit. A system that does use next fit will not go back and fill holes that had been freed until it reaches the end of memory. First fit will fill hole that appears first in memory that is large enough. Here’s proof of our changes going back to fill holes that had been freed. How we demonstrate what our allocation methods are doing is by printing out the free\_page\_bitmap surrounding the address that we allocated to. We print out 128 bits before, a pipe, and then 127 bits following the address. This printing is done every time that alloc\_pages is called. Additionally, we print out the number of pages requested as well as the page address, in hex, that we allocated to.



In this case the address 8da had been filled, but memory starting at 797 had been freed. Our changes went back and began to allocate memory in those freed addresses rather than continuing on from 8da as next fit would have done.

Another example of this is the following:



This example was taken after calling forkmem with 40MiB of allocation. If Minix was still using next fit, it would have allocated the last allocation on the screen to 921b, however, since some memory had been freed with the exiting of forkmem, it went back to 4e7a to allocate memory.

Additionally, we are submitting a couple of videos that show Minix running with our modified alloc.c as well as our print statements. The pictures included in this document were frames within the videos submitted.