ECE 2230 MP4
TEST PLAN
lab4.c
Cameron L. Burroughs

Unit Driver 1:

This driver is one provided to test general functionality of the Mem_alloc() and Mem_free() functions. This program ensures that all blocks are correctly allocated in the list and freed without gaps in memory. This driver also demonstrates a series of allocations that result in an empty free list (see red circle below). In the final memory list print, each block, with the exception of the dummy block, has an "end" location on another block that matches the "p" front location of the memory block. This verifies that all blocks, when inserted back into the free list, are still part of a continuous section of memory. See driver output below.

```
First-fit search policy without coalescing

----- Begin unit driver 1 -----
There are 256 units per page, and one unit is 16 bytes
first: 496 bytes (31 units) at p=0x1053e10
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x1053000 <-- dummy
p=0x1053000, size=224 (units), end=0x1053e00, next=0x6070e0

second: 2032 bytes (127 units) at p=0x1053610
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x1053000 <-- dummy
p=0x1053000, size=96 (units), end=0x1053600, next=0x6070e0

third: 1520 bytes (95 units) at p=0x1053010
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x6070e0 <-- dummy

unit driver 1: above Mem_print shows empty free list
fourth: 1008 bytes (63 units) at p=0x1054c10
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x1054000 <-- dummy
p=0x1054000, size=192 (units), end=0x1054c00, next=0x6070e0

first free of 1/8 a page p=0x1053e10
p=0x1053e00, size=32 (units), end=0x1054000, next=0x1054000
p=0x1054000, size=192 (units), end=0x1054c00, next=0x6070e0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x1053e00 <-- dummy

second free of 3/8 a page p=0x1053010
p=0x1053000, size=96 (units), end=0x1053600, next=0x1054000
p=0x1054000, size=192 (units), end=0x1054c00, next=0x6070e0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x1053e00 <-- dummy
p=0x1053e00, size=32 (units), end=0x1054000, next=0x1053000

third free of 1/2 a page p=0x1053610
p=0x1053600, size=128 (units), end=0x1053e00, next=0x1054000
p=0x1054000, size=192 (units), end=0x1054c00, next=0x6070e0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x1053e00 <-- dummy
p=0x1053e00, size=32 (units), end=0x1054000, next=0x1053000
p=0x1053000, size=96 (units), end=0x1053600, next=0x1053600

fourth free of 1/4 a page p=0x1054c10
unit driver 1 has returned all memory to free list
p=0x1054c00, size=64 (units), end=0x1055000, next=0x1054000
p=0x1054000, size=192 (units), end=0x1054c00, next=0x6070e0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x1053e00 <-- dummy
p=0x1053e00, size=32 (units), end=0x1054000, next=0x1053000
p=0x1053000, size=96 (units), end=0x1053600, next=0x1053600
p=0x1053600, size=128 (units), end=0x1053e00, next=0x1054c00

~~~~~~~~~~ FREE LIST MEMORY STATS ~~~~~~~~~~
Items in free list (dummy block included): 6
Minimum Block Size: 512 bytes
Maximum Block Size: 3072 bytes
Average Block Size: 1632 bytes
Total Memory: 8192 bytes
Number of calls to sbrk(): 2
Total Pages: 2
all memory is in the heap -- no leaks are possible
```

Unit Driver 2:

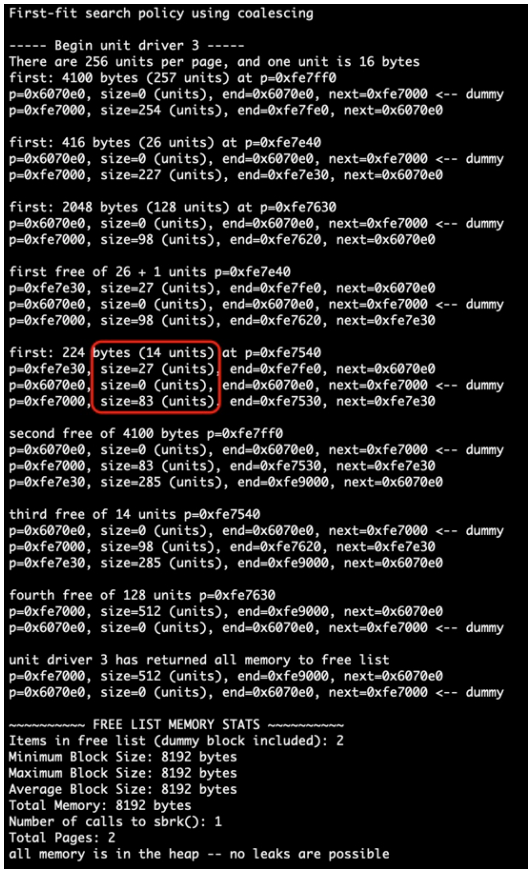
Student-made driver to test bounds of adding new pages to the free list. This driver tests three conditions: (1) Requesting exactly the number of bytes in a page (256 units), (2) Requesting one unit more than the number of bytes in a page (257 units), and (3) requesting one unit less than the number of bytes in a page (255 units). The expected output for the first request of 256 units would be two pages of memory requested with 257 units being returned to the user (after the header is added), and 255 units left in the free list. For the second request, again, two pages should be requested, but this time, 258 units would be returned to the user and an additional 254 units will be left in the free list. Finally, the third request will result in one page

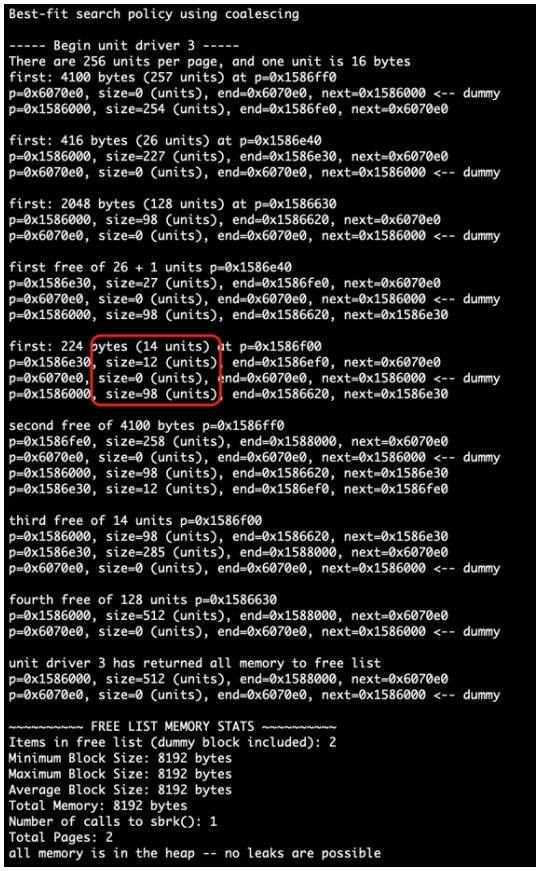
being returned to user (256 bytes) and no additional memory added to the free list. See output below.

```
----- Begin unit driver 2 -----
There are 256 units per page, and one unit is 16 bytes
first: 4096 bytes (256 units) at p=0x1476000
p=0x6060e0, size=0 (units), end=0x6060e0, next=0x1475000 <-- dummy
p=0x1475000, size=255 (units), end=0x1475ff0, next=0x6060e0

first: 4112 bytes (257 units) at p=0x1477ff0
p=0x1475000, size=255 (units), end=0x1475ff0, next=0x1477000
p=0x1477000, size=254 (units), end=0x1477fe0, next=0x6060e0
p=0x6060e0, size=0 (units), end=0x6060e0, next=0x1475000 <-- dummy

first: 4080 bytes (255 units) at p=0x1479010
p=0x1477000, size=254 (units), end=0x1477fe0, next=0x6060e0
p=0x6060e0, size=0 (units), end=0x6060e0, next=0x1475000 <-- dummy
p=0x1475000, size=255 (units), end=0x1475ff0, next=0x1477000

first free of 258 units p=0x1476000
p=0x1475000, size=766 (units), end=0x1477fe0, next=0x6060e0
p=0x6060e0, size=0 (units), end=0x6060e0, next=0x1475000 <-- dummy

first free of 257 units p=0x1477ff0
p=0x6060e0, size=0 (units), end=0x6060e0, next=0x1475000 <-- dummy
p=0x1475000, size=1024 (units), end=0x1479000, next=0x6060e0

first free of 256 units p=0x1479010
p=0x6060e0, size=0 (units), end=0x6060e0, next=0x1475000 <-- dummy
p=0x1475000, size=1280 (units), end=0x147a000, next=0x6060e0

unit driver 2 has returned all memory to free list
p=0x6060e0, size=0 (units), end=0x6060e0, next=0x1475000 <-- dummy
p=0x1475000, size=1280 (units), end=0x147a000, next=0x6060e0
```

Unit Driver 3:

This driver tests that the program will correctly allocate enough space for a requested byte amount that is not a multiple of 1 unit, sizeof(mem_chunk_t). This program also tests best fit vs first fit. First, 4100 bytes are requested. This would be 256.25 units, so the program rounds up to 257 units, then adds one more unit (the header) for a total of 258 requested units. The result is two pages requested through one call to sbrk(). 258 units will be returned to the user while 254 will remain in the free list. Next, two blocks of 26 units and 128 units are requested, then the block of 26 units is freed, putting 27 units back in the free list. This order allows there to be a gap between the remaining 98 units on the free list and the newly freed 27 units. This gap is what will allow us to test first fit vs best fit regardless of the state of coalescing. At this point, two blocks will be present, 98 units and 27 units, and the rover will be on the block of 98 units. Next, 14 units is requested from the free list. If first fit is on, the 14 units should be taken from the 98-unit block, however best fit should take the 14 units from the 27-unit block. Finally, all memory is freed, and the test is concluded. See output below for results of first fit vs best fit.

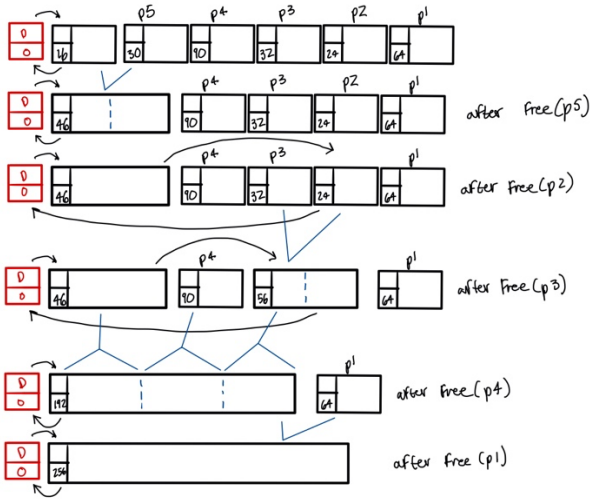




Unit Driver 4:

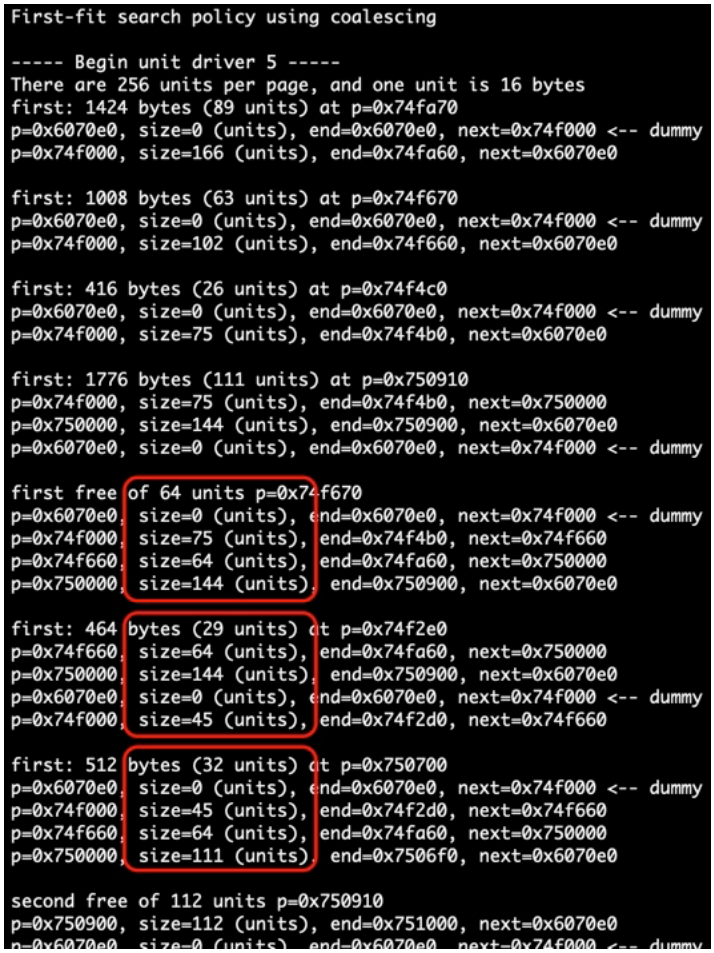
This driver tests that the program will correctly use coalescing in various special conditions. These special conditions include coalescing a memory block that is directly in front of another, coalescing a memory block that is directly behind another, and coalescing a memory block that is directly between two existing memory blocks on the free list. First, when p5 is freed, the driver demonstrates its ability to combine the freed pointer with a block it is contiguous to on the front end by combining with the 16-unit block in front of it. When p3 is freed, the programs ability to combine the freed pointer with a block it is contiguous with on the back end as it combines with the previously freed p2 pointer (24 units). When p4 is freed, the program demonstrates its ability to combine the freed pointer with two blocks it is contiguous with on the front and back ends at the same time (the two previously combined blocks, totaling 192 units). Finally, when p1 is freed, the programs ability to combine a continuous block on the wrap end of the free list is demonstrated as it combines with the rest of the list into one contiguous block of memory. See the figure below to visualize this test. The output of the driver is also seen below.

```
First-fit search policy using coalescing

----- Begin unit driver 4 -----
There are 256 units per page, and one unit is 16 bytes
first: 1008 bytes (63 units) at p=0xe49c10
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=192 (units), end=0xe49c00, next=0x6070e0

first: 368 bytes (23 units) at p=0xe49a90
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=168 (units), end=0xe49a80, next=0x6070e0

first: 496 bytes (31 units) at p=0xe49890
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=136 (units), end=0xe49880, next=0x6070e0

first: 1424 bytes (89 units) at p=0xe492f0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=46 (units), end=0xe492e0, next=0x6070e0

first: 464 bytes (29 units) at p=0xe49110
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=16 (units), end=0xe49100, next=0x6070e0

first free of 30 units p=0xe49110
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=46 (units), end=0xe492e0, next=0x6070e0

second free of 24 units p=0xe49a90
p=0xe49a80, size=24 (units), end=0xe49c00, next=0x6070e0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=46 (units), end=0xe492e0, next=0xe49a80

third free of 32 units p=0xe49890
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=46 (units), end=0xe492e0, next=0xe49880
p=0xe49880, size=56 (units), end=0xe49c00, next=0x6070e0

fourth free of 90 units p=0xe492f0
p=0xe49000, size=192 (units), end=0xe49c00, next=0x6070e0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy

fifth free of 64 units p=0xe49c10
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=256 (units), end=0xe4a000, next=0x6070e0

unit driver 4 has returned all memory to free list
p=0x6070e0, size=0 (units), end=0x6070e0, next=0xe49000 <-- dummy
p=0xe49000, size=256 (units), end=0xe4a000, next=0x6070e0

~~~~~~~~~~ FREE LIST MEMORY STATS ~~~~~~~~~~
Items in free list (dummy block included): 2
Minimum Block Size: 4096 bytes
Maximum Block Size: 4096 bytes
Average Block Size: 4096 bytes
Total Memory: 4096 bytes
Number of calls to sbrk(): 1
Total Pages: 1
all memory is in the heap -- no leaks are possible
```



Unit Driver 5:

This driver demonstrates that the rover moves after each allocation and free, therefore it does not cluster fragments of memory at the beginning of the free list. First, a series of allocations and frees are preformed such that a list of 3 non-contiguous blocks and the dummy block are present with exactly 75, 64, and 144 units free. Next, a block of 30 units is requested, leaving the list with three blocks with 45, 64, and 144 units free. It is clear that the 30 units were taken from the block with 75 units originally. Next, a block of 33 units is requested. Each remaining block in the list has enough space to accommodate this request, however since the rover moves along the list, the 33 units is taken from a different block than the original 75-unit block. While the rover transverses the list, the blocks get broken down at a relatively even distribution such that no small memory fragments are bunched at the beginning of the list. See the output of this test below.


First-fit search policy using coalescing
----- Begin unit driver 5 -----
There are 256 units per page, and one unit is 16 bytes
first: 1424 bytes (89 units) at p=0x74fa70
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x74f000 <-- dummy
p=0x74f000, size=166 (units), end=0x74fa60, next=0x6070e0
first: 1008 bytes (63 units) at p=0x74f670
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x74f000 <-- dummy
p=0x74f000, size=102 (units), end=0x74f660, next=0x6070e0
first: 416 bytes (26 units) at p=0x74f4c0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x74f000 <-- dummy
p=0x74f000, size=75 (units), end=0x74f4b0, next=0x6070e0
first: 1776 bytes (111 units) at p=0x750910
p=0x74f000, size=75 (units), end=0x74f4b0, next=0x750000
p=0x750000, size=144 (units), end=0x750900, next=0x6070e0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x74f000 <-- dummy
first free of 64 units p=0x74f670
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x74f000 <-- dummy
p=0x74f000, size=75 (units), end=0x74f4b0, next=0x74f660
p=0x74f660, size=64 (units), end=0x74fa60, next=0x750000
p=0x750000, size=144 (units), end=0x750900, next=0x6070e0
first: 464 bytes (29 units) at p=0x74f2e0
p=0x74f660, size=64 (units), end=0x74fa60, next=0x750000
p=0x750000, size=144 (units), end=0x750900, next=0x6070e0
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x74f000 <-- dummy
p=0x74f000, size=45 (units), end=0x74f2d0, next=0x74f660
first: 512 bytes (32 units) at p=0x750700
p=0x6070e0, size=0 (units), end=0x6070e0, next=0x74f000 <-- dummy
p=0x74f000, size=45 (units), end=0x74f2d0, next=0x74f660
p=0x74f660, size=64 (units), end=0x74fa60, next=0x750000
p=0x750000, size=111 (units), end=0x7506f0, next=0x6070e0
second free of 112 units p=0x750910
p=0x750900, size=112 (units), end=0x751000, next=0x6070e0