VPOS's MMU Theory

Author: Benjamin Gehant

Contributions:

- Ryan DeBarger for getting rid of the in-use byte, using bit merging.

MMU Abstract:

Malloc's "first stage" is to fill all of heap memory with blocks of varying sizes and ID's. After heap memory has been filled, malloc will switch to the "second stage" and will permanently loop through a linked list of free blocks constructed using the free function in-order to find free memory.

Heap Structure:

heap_ptr
head_ptr
tail_ptr
mem-start
block1
block2
block3
blockn
mem-end

Block Structure:

block size (14 bits)	block id(2 bits)	→ 2 bytes
data (32 + x bits)		→ 4 + x bytes
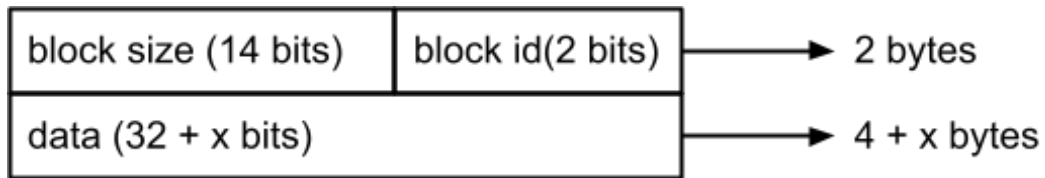
Minimum block size: 4 bytes
Max block size: 16383 bytes

Overhead:
$$x < 4 \Rightarrow f(x) = (4 - x) + 2$$
$$x \geq 4 \Rightarrow f(x) = 2$$
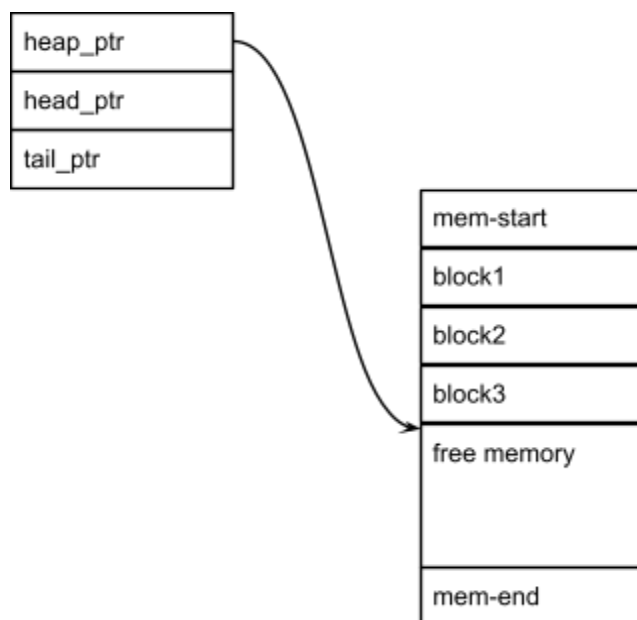
ID's:
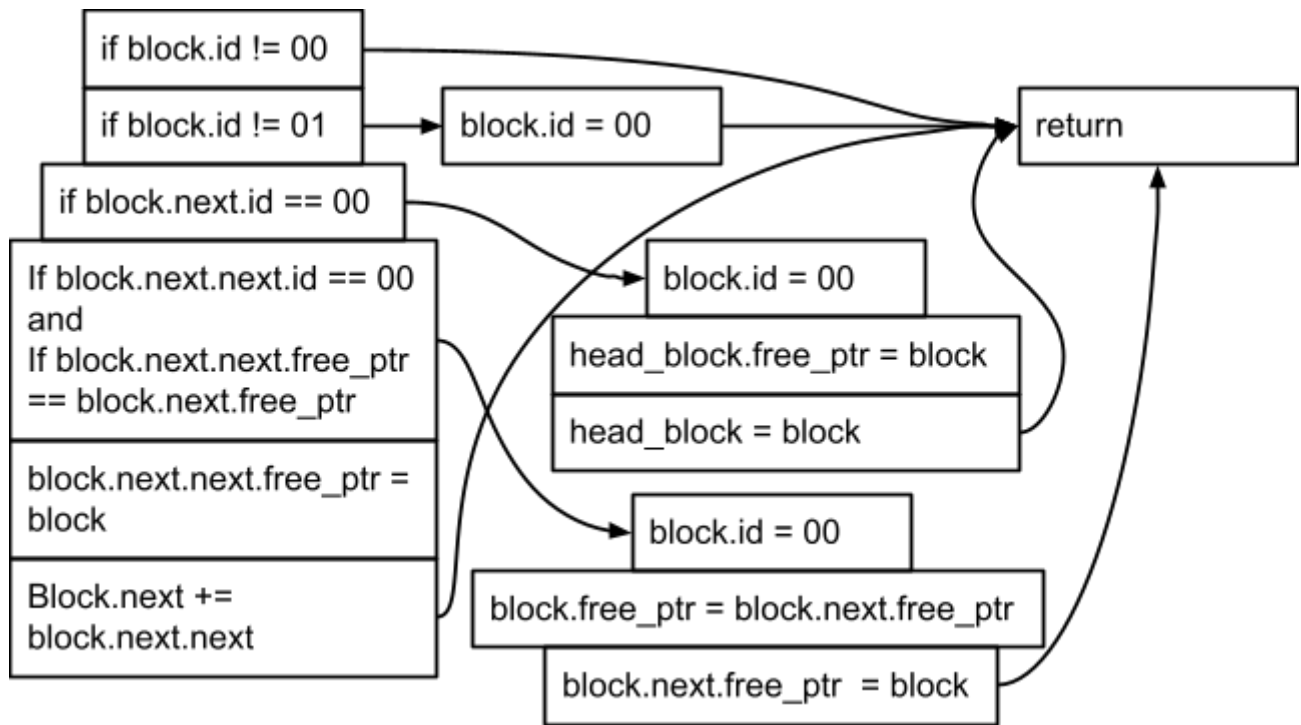   11 - allocated block
    01 - there is a previous free block
    00 - free block

Malloc First Stage:

    The heap pointer points to free heap memory and places a block there until there's no free memory.

heap_ptr
head_ptr
tail_ptr

mem-start
block1
block2
block3
free memory
mem-end

Free:

```
if block.id != 00

if block.id != 01          block.id = 00                              return

if block.next.id == 00

If block.next.next.id == 00              block.id = 00
and
If block.next.next.free_ptr              head_block.free_ptr = block
== block.next.free_ptr
                                         head_block = block
block.next.next.free_ptr =
block
                                         block.id = 00
Block.next +=
block.next.next              block.free_ptr = block.next.free_ptr

                             block.next.free_ptr  = block
```

Probably got some semantics wrong