Recitation6: Malloc Lab

CSE251, Spring 2019
Sehoon Kim
sshhee@unist.ac.kr

Lab info

- Due
  - May 28 (Tue), 11:59PM
- TA’s
  - Sehoon Kim (sshhee@unist.ac.kr, Tue 14:30~15:30 @106-605)
  - Anvar Alisheri (alisher@unist.ac.kr, Thu 19:30~20:30 @106-709)

Malloc Lab

- What to do?
  - Make `mm_init`, `mm_malloc`, `mm_free` and `mm_realloc`.
  - Grade will be measured by correctness and performance.

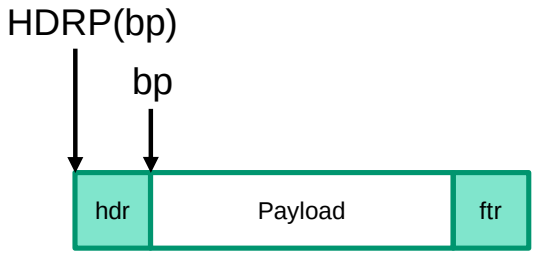
Malloc Lab – Guide Line

- Start point
  - You are encouraged to start from simple implicit free list (textbook 883-897p).
  - First, fully understand the concept and implementation of implicit free list.
  - Complete the implementation of implicit free list by writing missing parts.

Malloc Lab – Guide Line

- Macros (893p)
  - Encapsulate your pointer arithmetic in C preprocessor macros.

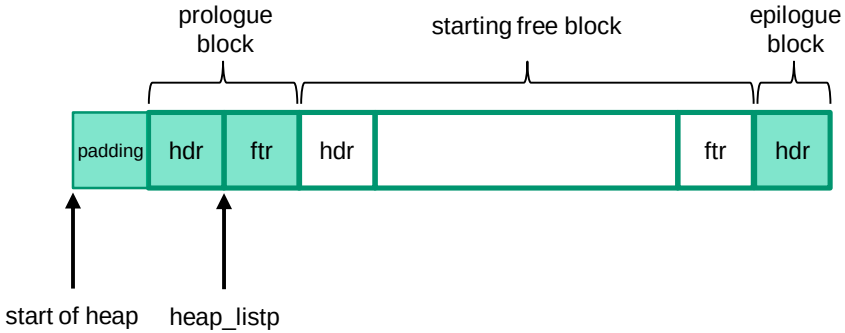
ex) `#define HDRP(bp)  ((char*)(bp) - WSIZE)`



You can easily get an address of header that must be used for memory operations.

Malloc Lab – Guide Line

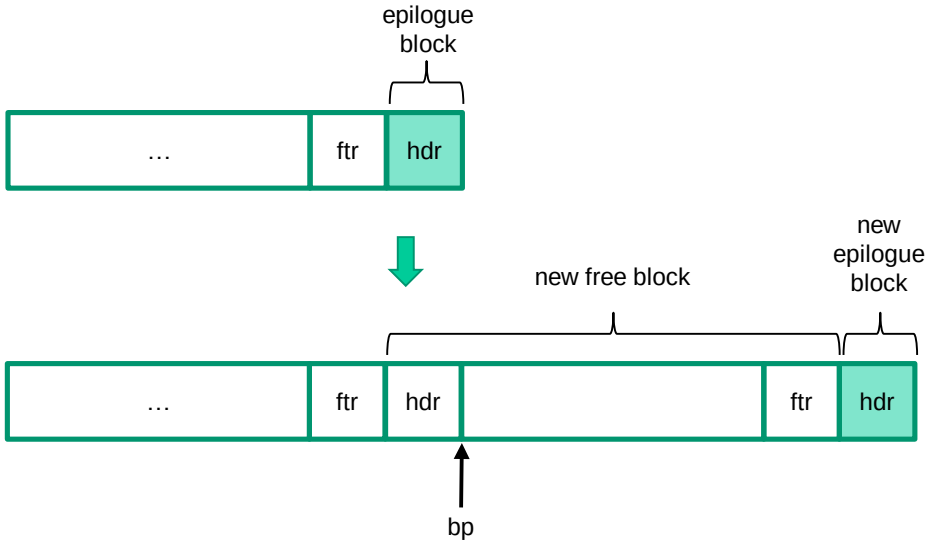
- `int mm_init(void)`
  - Creates and initializes heap.



NOTE: You must declare a global variable `static char *heap_listp` beforehand.

Malloc Lab – Guide Line

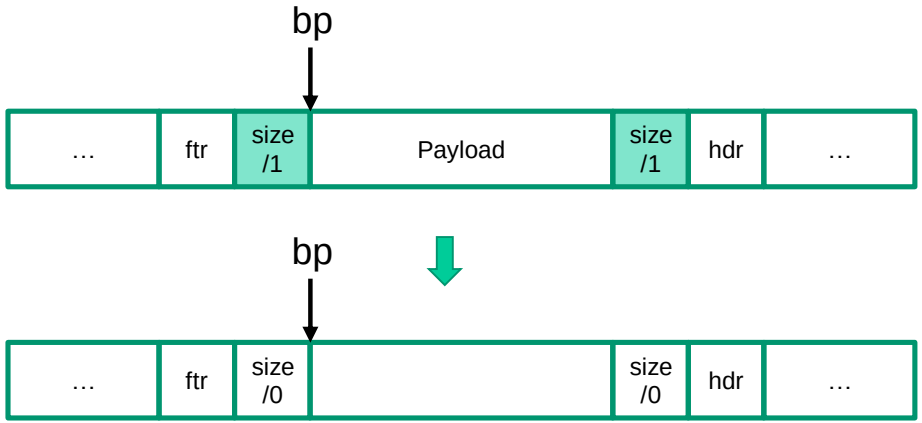
- `static void *extend_heap(size_t words)`
  - Entends the heap with a new free block.



then, call `coalesce(bp)`

Malloc Lab – Guide Line

- `void mm_free(void *bp)`
  - Unset allocated bit and coalesces.

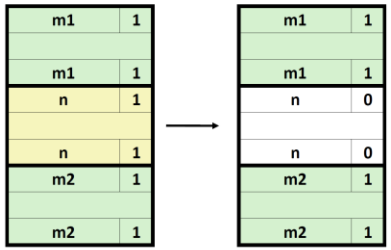


then, call `coalesce(bp)`

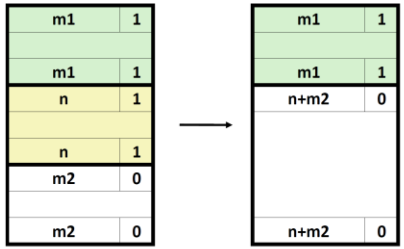
Malloc Lab – Guide Line

- `static void *coalesce(void *bp)`

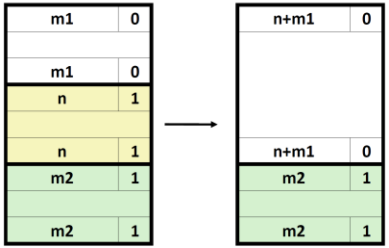
Case 1



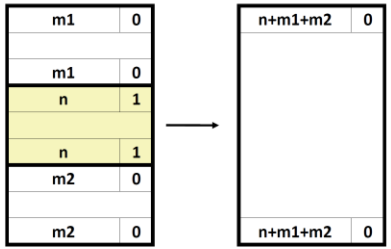
Case 2



Case 3



Case 4



Malloc Lab – Guide Line

- `void *mm_malloc(size_t size)`
  - First, find a proper space by `find_fit()`.
    If there are no room, `extend_heap()`.
  - Then, allocate blocks by `place()`.

Malloc Lab – Guide Line You have to implement!

- `static void *find_fit(size_t asize)`
  - It must find a space that is able to accommodate `asize` within the heap.
  - If search succeeded, return its block(payload) address, not header address.
    Else search failed, return NULL.
  - You can use any policy.
    (For example, first fit, next fit, best fit….)

Malloc Lab – Guide Line You have to implement!

- `static void place(void *bp, size_t asize)`
  - If the size of the remainder would equal or exceed the minimum block size(header+footer+1word+alignment padding), splits.
  - Allocates block to `bp` by setting header and footer. If remainder exists, sets their header and footer.

Malloc Lab – Guide Line

- Now, your implicit free list will work.
  - You still need to make your `realloc()`.
  - You may not be satisfied with the performance.
    If then, try other allocators!

Malloc Lab – Notice

- Again, understand simple implicit free list and read descriptions carefully before starting assignment.
- In description, there are programming rules you must observe.
  Do not violate them.
- You have to work on uni06 server
  because trace files are available only on uni06 server.

Malloc Lab – Hints

- Trace files are in `/data/traces` of uni06 server.
  `./mdriver` will automatically load them as the path is involved in `config.h`.

- Q: Is it allowed to make use of codes in textbook?
  A: yes.

- Codes on 891p are not supposed to be made by you.
  They are implemented in `memlib.c`.
  You can just call them.