

Homework 1

Problem 1

In the simple allocator example shown at the section 9.9.12, why there is one **prologue block**? Give out your reason. (Hint: See the code of function *prev_alloc*)

Problem 2

Implement a find best fit function for the simple allocator in the book. it should perform a best fit search of the implicit free list. Write down your code. (if needed, add some comments)

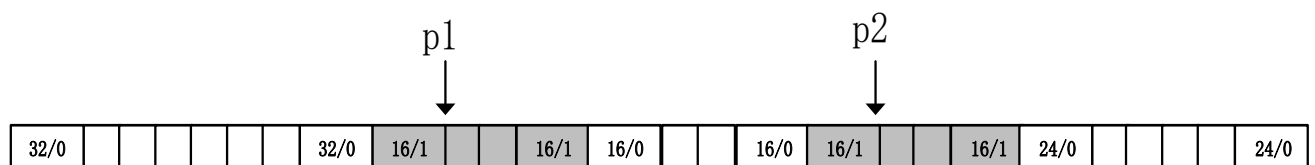
static void *find_best_fit (size_t asize)

Problem 3

What is a false fragmentation? And how to solve this phenomenon? Please list at least two ways and make a simple introduction about them.

Problem 4

The figure simulates the initial status of memory at a certain time. Allocated blocks are shaded, and free blocks are blank (each block represents 4 bytes). The allocator maintains double-word alignment. Given the execution sequence of memory allocation operations (*malloc()* or *free()*) from 1 to 4.



- 1: *P3 = malloc(4);*
- 2: *P4 = malloc(10);*
- 3: *free(P1);*
- 4: *P5 = malloc(16);*

Assume **first-fit** algorithm is used to find free blocks and coalesce immediately. Please draw the status of memory and mark with variables after the **2nd** and **4th** operation is executed.