Homework 12

1. Determine the block sizes and header values that would result from the following sequence of malloc requests.

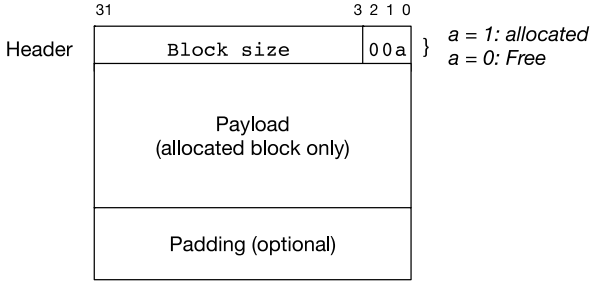
Request	Block size (decimal byte)	Block header (hex)
`malloc(1)`		
`malloc(5)`		
`malloc(12)`		
`malloc(13)`		

Assumptions:

a) words are 4-byte objects

b) The memory allocated to the user is at the granularity of word. That is, the size requested are rounded up to the nearest multiple of 4 bytes.

c) The allocator maintains double-word alignment and uses an implicit free list with the block format as below.

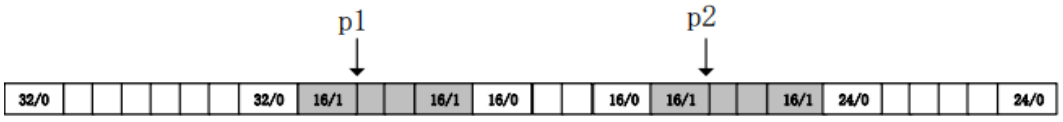


2. Determine the minimum block size for each of the following combinations of alignment requirements and block formats. Assumptions: implicit free list, zero-size payloads are not allowed, and headers and footers are stored in 4-byte words.

Alignment	Allocated block	Free block	Minimum block size (bytes)
Single word	Header and footer	Header and footer	
Single word	Header, no footer	Header and footer	
Double word	Header and footer	Header and footer	
Double word	Header, no footer	Header and footer	

Exercise 9

1. The figure simulates the initial status of memory at a certain time. Allocate blocks are shaded, and free blocks are blank (each block represents 4 bytes). The allocator maintains **double-word alignment**. Assume that **immediate coalescing** strategy and **splitting free blocks** are employed.



Given the execution sequence of memory allocation operations:

```
1: P3 = malloc(4);
```

```
2: P4 = malloc(10);
3: free (P1);
4: P5 = malloc(16);
```

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

b) Assume **best-fit** algorithm is used to find free blocks. Please draw the final status of memory and mark with pointers. And calculate the total bytes of the internal fragments. (NOTE: NO need to consider p1 and p2 when calculating internal fragments)

2. Suppose there's a program consisting of 3 parts A, B & C. Part A takes up 20% of the overall run time, while part B takes up 30%, part C 50%. You decide to optimize one of the three parts to improve the performance. For the same cost, you can speed up A by a factor of 3.0, or speed up B by a factor of 2.5, or C by a factor of 1.5. Which choice would maximize the performance? And please give the speedup of them.

3. Write HCL code describing a circuit that for word inputs A, B, and C selects the *median* of the three values. That is, the output equals the word lying between the minimum and maximum of the three inputs.