Report of Homework 1 – ECE650

Section 1: Requirements and summary of development.

First Fit and Best Fit malloc are memory allocation algorithms used in Operating Systems to allocate memory to processes. The main objective of these algorithms is to manage memory efficiently by allocating blocks of memory to processes in an efficient manner.

The main requirements of these algorithms are to minimize fragmentation, maximize utilization of available memory, and allocate memory blocks in a way that meets the requirements of the processes.

The environment used for the development of these algorithms is ubuntu 18.04. The tools used include vim, gdb, and vscode.

Section 2: Design, implementation and test

Data structure:

I insert an informational block at both the beginning and end of each block. The informational block retains the length of the present block and its validity status. The blocks are continuous, thus enabling navigation to the next and previous blocks through the length of the current block. By this way I implement a doubly linked list to connect each block.

Functions:

First fit malloc:

1. Malloc: The malloc function will either allocate a new block using sbrk() if no block has been allocated, or it will iterate through the memory blocks, checking if any are valid and have enough size to fulfill the request, and try to find the first block which meets the requirement. If it finds a suitable block, it will attempt to split it if possible. If no usable block is found, it will again call sbrk() to allocate a new block.

2. free: The free function first resets the memory block and marks it as invalid, then attempts to merge it with the adjacent blocks. It also keeps track of the first valid block, which is used as a starting point for iteration when allocating new blocks with malloc

Best fit malloc:

1. Malloc: The malloc function allocates a new block with sbrk() if no suitable block is found. It searches memory, records the address of the smallest suitable block and splits it if possible. If no block is found, it will also call sbrk().

2. free: The free function first resets the memory block and marks it as invalid, then attempts to merge it with the adjacent blocks. It also keeps track of the first valid block, which is used as a starting point for iteration when allocating new blocks with malloc.

Difference:

In First fit, the malloc function finds the first suitable block and allocates it. In Best fit, it finds the smallest suitable block and allocates it. The objective of First fit is to minimize fragmentation, while the objective of Best fit is to minimize waste.

Similarities:

Both algorithms keep track of the first valid block for use as a starting point when allocating new blocks.

Testing: I use the general test cases to prove the correctness of my code and use the alloc policy test to test the speed and the quality of the code.

Section 3: Performance Results & Analysis

The following sheet shows the performance of these two methods.

FF	fragmentation	Execution time	BF	Fragmentation	Execution Time
Small_range_rand_allocs	0.97	92.1	Small_range_rand_allocs	0.99	72.7
Large_range_rand_allocs	0.98	103.1	Large_range_rand_allocs	0.99	249.9
Equal_size_allocs	0.99	496	Equal_size_allocs	0.99	503.5

Difference between first-fit and best-fit malloc in the tests:

1.equal_size_allocs:

Fragmentation: In this test, both first-fit and best-fit malloc will have high fragmentation as the allocation size is fixed and freed in the same order as created so there isn’t any block being merged.

Execution time: Both algorithms will have similar execution time as the allocation size is fixed and does not require iterating through memory to find the best-fitting block.

2.small_range_rand_allocs:

Fragmentation: In this test, best-fit malloc will result in higher fragmentation compared to first-fit malloc, as best-fit allocates the smallest block that fits the request.

Execution time: Best-fit malloc may have longer execution time as it iterates through the entire memory block to find the smallest suitable block, whereas first-fit malloc has a faster execution time as it finds the first block that fits the requested size.

3.large_range_rand_allocs:

Fragmentation: In this test, best-fit malloc will result in higher fragmentation compared to first-fit malloc, as best-fit allocates the smallest block that fits the request.

Execution time: The execution time of best-fit malloc may be significantly longer than first-fit malloc in this test, as the allocation size is much larger and best-fit needs to iterate through the entire memory block to find the best fit.