Lab #4: Heap Management

Overview

The purpose of this lab is for you to explore some aspects of memory allocation and heap management.

A simple implementation of `malloc` is given in Section 8.7 (page 185) of the "C Programming Language" (you can download the source files from the course website). This implementation uses a "first fit" algorithm when selecting which free block to allocate from.

For this lab you will modify the given implementation to change the allocation algorithm. The purpose is to gain an understanding of how `malloc` and `free` work.

Part 1: `best fit`

Modify the given code to implement a "best fit" strategy (store this modification in a file called `best.c`). Under such a strategy the free list is searched for the smallest available block that will satisfy the request. You should only request additional memory from the operating system (using `sbrk`, as is already done) if there is no block large enough to satisfy the request.

Part 2: `worst fit`

Modify the (original) given code for `malloc` to implement a "worst fit" strategy (store this modification in a file called `worst.c`). Under such a strategy the free list is searched for the largest available block. You should only request additional memory from the operating system (using `sbrk`, as is already done) if there is no block large enough to satisfy the request.

Part 3: `main`

A `main` function has been provided for you to use in demonstrating that your allocators work. Run each program generated by make to see where each allocated block is placed. Draw pictures to help with your understanding.

Demonstration

Once you are convinced that your allocators work properly and you understand how `malloc` and `free` work in this implementation, demonstrate the working program to the instructor to receive credit for this lab.