

First fit

I would use a linked list to apply the free space management strategy. There are 2 fields in the linked list struct:

```
struct freeSpace {  
    int address;  
    struct freeSpace* next;  
}
```

I would create a memory size of 1000. Based on the first fit principle, once the program finds an appropriate chunk, it would allocate space for the task. Once there are contiguous chunks, they would merge into one piece automatically.

The program would print the final state of the free space.

Best fit

Iterate through the linked list and keep the current chunk size and address in a temporary variable. If there is a chunk found that satisfies the new request and the chunk size is smaller than the previously stored chunk size, store this chunk in the temporary variable. Iterate through the linked list until it reaches the end. Return the chunk that was temporarily stored.

Next fit

Based on the first fit, add a ptr "last" to trace the last visited address. And traverse through the rest of the linked list until we find the first space that could fit the mem request. Then change the ptr last to the curr visit and updated the free list, return the chunk.

