

Major Program 3: Memory Manager

Due: Friday March 31 at 11:59 pm

The purpose of this assignment is to give you experience working with memory management algorithms.

You must submit a module called memmgr.c which has 3 functions:

```
/**/
/* This function is called ONE TIME ONLY.  It uses      */
/* regular malloc(size) to allocate the block to be    */
/* managed.  It should also malloc a block of 100      */
/* PDE's put 99 of them on free PDE list and one of    */
/* them on the free partition list.                    */
*/

void xm_init(
int size)      /* Size of big gob of memory in bytes */

/**/
/* This entry is used to allocate a sub-gob from      */
/* the big gob.. It should return 0 if the amt        */
/* requested is not available. It should return        */
/* the address of the allocated memory if it is.      */
*/

void *xm_alloc(
int size)      /* Size of little gob wanted */

/**/
/* This entry is used to free a previously allocated  */
/* subgob.  You may assume no bogus addresses will   */
/* be passed in.                                     */
*/

void xm_free(
void *addr)    /* Address of block to be freed */
```

Your memory manager MUST use the approach described in class and in the notes.

There is a test driver available named ptmem.c in examples/ptthread

to compile:

```
gcc -g -DDEBUG memmgr.c ptmem.c -lpthread -lrt
```

or

```
gcc -g -DTHE_REAL_DEAL -DDEBUG memmgr.c ptmem.c -lpthread -lrt
```

or if you just want to verify that it works using real malloc

```
gcc -g -DTHE_REAL_DEAL -DUSE_MALLOC -DDEBUG memmgr.c ptmem.c  
-lpthread -lrt
```

How to submit your program:

NOTE: This procedure has NOTHING in common with "handin" nor "sendlab"
Do NOT even TRY to think about how they fit into this procedure because
THEY DON'T!!

<<<Do NOT turn in any core files, makefiles etc.>>>

You must turn in 1 file: memmgr.c

1. From any departmental Solaris system *ssh* to workstation *jmw*
2. The submission directories lie in the directory `/local/jmw2/322/mp3` which is available ONLY IF YOU HAVE LOGGED INTO WORKSTATION *jmw*. Each student has a subdirectory of `/local/jmw2/322/mp3`. The name of your subdirectory is your userid (in the example we will assume your id is *wjsmith*).
3. copy (via the `cp` command) required file to your subdirectory in `/local/jmw2/322/mp3`

For example:

```
cp /home/wjsmith/322/mp3/memmgr.c
/local/jmw2/322/mp3/wjsmith
```

Here you would (hopefully) obviously need to replace
`/home/wjsmith/322/mp3/memmgr.c`
with wherever you have your program.

4. Don't modify the permissions on your subdirectory. They are set so that ONLY you can access your directory.

After you think you have turned your programs in, its a good idea to
`cd /local/jmw2/322/mp3/wjsmith`
and make sure your files are there and they still compile and work correctly.