B4 - Unix System Programming

B-PSU-400

malloc

Memory ALLOCation

2.1

malloc

binary name:	libmy_malloc.so
group size:	2
repository name:	PSU_$ACADEMICYEAR_malloc
repository rights:	ramassage-tek
language:	C
compilation:	via Makefile, including re, clean and fclean rules

- Your repository must contain the totality of your source files, but no useless files (binary, temp files, obj files,...).
- All the bonus files (including a potential specific Makefile) should be in a directory named *bonus*.
- Error messages have to be written on the error output, and the program should then exit with the 84 error code (0 if there is no error).

Unauthorized Functions: mmap, munmap, *alloc, free, dlopen, dlsym.

Using only **brk/sbrk** and your amazing mind, rewrite the 3 following C library functions: **malloc**, **free** and **realloc**.

Do not even think about using the malloc function.

Your function prototypes must be the same as the ones from the C library, and must be contained in the library compiled by your Makefile.

```
void *malloc(size_t size);
void *realloc(void *ptr, size_t size);
void free(void *ptr);
```

All programs using malloc should work with your own malloc. Remember to test it with several existing programs.
Some programs use their own allocation system and memory management. Think about what kind of tests you're going to perform.

In order to give you more control, also write a **show_alloc_mem()** function.
It displays the status of the allocated areas on the screen (to be inserted into the library).
The display must look like the following, with the addresses in ascending order:

```
break : 0xB0000
0xA0000 - 0xA41CA : 16842 bytes
0xAE000 - 0xAE03F : 63 bytes
....
```

A bot is available on the intra to test your functions; it will be used during evaluation.

Do not limit yourself by exclusively testing with the bot. It only checks simples cases... More advanced tests will be done during evaluation(small and large allocations, series of malloc/free, RAM optimization...).

Addresses returned by malloc (and others) are aligned.

As a bonus, you could handle */etc/malloc.conf* file.