Assignment 1: Fine Techniques

Marks: 15

**Submission Date: 9/03/2019, EOD**

Mode of Submission: Google Form - Each student will be required to submit one implementation (.c) file and a README with instructions on how to compile their implementation file as well as any additional information.

Please note that the client file will not be required.

The link to the Google Form shall be provided later.

The form will close at 12:01 am, and no late submissions as emails shall be considered.

1. Use an array of char of size n. The value of n shall be provided as input as a parameter to the required functions.

2. Attached with this email is a header file with definitions for 5 functions.

void allocate(int n)

(void\*) mymalloc(int size)

void myfree (void \*)

void display\_mem\_map()

void print\_free\_list()

* the allocate function allocates memory for the char array of size n, which will be global to all the other functions.
* display\_mem\_map will print a view of your memory at any instance that it is called.

For every block, print the starting address, ending address and a boolean value (0 if it has not been allocated, 1 if it has been allocated).

* print\_free\_list prints the starting address and block size of all the free blocks in memory, newline separated, at the time when the function is called.

[refer to the header file for sample outputs]

3. Please ensure that if you are using any helper functions in your implementation file, they are named with the static keyword so they don't clash with any functions that we will use to test your code.

4. Each student will implement one allocation policy. Following are the list of allocation policies:

1. first fit

2. best fit

3. worst fit

**(class roll number) % 3+1 will be the allocation policy you will be implementing.**

5. Codes submited with compilation errors will directly be rejected.

6. In addition to the header file provided to you, please ensure that you only include header files in gcc's standard collection.

7. Before the submission date, a sanity client along with the expected output for each of the allocated methods will be sent to you. Please make sure your code generates the expected output for the sanity client before submitting it.

8. Writing comments where necessary, using meaningful variable names, making sure that all your functions have a single point of return and well-indented code will carry marks as well.

9. Any situations that involve plagiarism will be awarded 0 marks.