

# 91.308 Operating Systems Spring 2005

## Project 6 – Memory Management Due Friday, April 1

- This assignment is due on Fri, Apr 1.
- You must submit the **course cover-sheet**, **hardcopy** of your source code, your output and a short write-up described below. The problem you must solve has been described in class and is formalized as follows:
- The program you will write will read a file which contains memory allocation requests and memory free operations. The allocation and free operations will be made against an initially empty fixed sized free memory pool. The program must support three allocation policies:
  - First fit linked list
  - Best fit linked list
  - Buddy system power of 2 block allocation (using a minimum allocation of 32 bytes)
- Your program must:
  1. Accept three command line arguments, such that argv[1] is the **policy** to use, argv[2] is the **total memory free pool size** and argv[3] is the **name of the file** containing the allocation and free requests. The memory size of the total free pool for the main test case will be **1 MByte**. You may use other sizes in any additional experiments you attempt, but make sure you **describe** your configuration and results in your write-up.
  2. Read each line of the input file in argv[3] and carry out one of the following actions:
    - Make a memory allocation if enough memory is available to satisfy an allocation request.
    - Return memory to a linked list and carry out any coalescing (buddying up) required.
    - Refuse the allocation only if there is not enough memory to satisfy it in any available free partition. Refused requests are discarded and never considered again, so make sure you discard the corresponding free operation when you see it in the input stream.
  3. Generate one line of output for each line of input, providing information about which request this is, how the request was handled, what the total amount of free space after the request is, and what the size

of the largest free partition after the request is. Sample input and output files are shown below.

**INPUT FORMAT:**

SERIAL-NUM	REQUEST	SIZE	SERIAL-MATCH
1	alloc	20000	
2	alloc	100000	
3	alloc	5050	
4	free		2
5	alloc	70500	
6	free		1
7	alloc	400000	

**OUTPUT FORMAT:**

MANAGEMENT POLICY = First Fit POOL SIZE = 512 KB

SERIAL-NUM	REQUEST	SIZE	ALLOC-ADDR	TOTAL-FREE	LARGEST-PART
1	alloc	20000	0 504288	504288	
2	alloc	100000	20000 404288	404288	
3	alloc	5050	120000 399238	399238	
4	free	100000	499238	399238	
5	alloc	70500	20000 428738	399238	
6	free	20000	0 448738	399238	
7	alloc	400000	-1 448738	399238	

- Your output for submission should include:
  - The course cover-sheet, with path names to your source and executable for possible testing, and a statement of your success level as in other assignments.
  - A hard copy of your source code.
  - A hard copy of the first 3 pages and the last 3 pages of each of your output files (one file per policy) for the main test case (**1 MB memory**).
  - A brief write-up which should include a short discussion of your results and any observations or problems you encountered in the project.
- The data you must use for this project will be available in **~fredm/308/files/proj6\_data** or at the class website.

**Credits: This assignment was developed by Prof. Moloney, UML CS Dept.**