CS4375 Fall 2025 Homework Report

<Your name> Submitted on <date>

<Your email address>

**HW 4: Lazy Allocation for xv6**

Please replace red text with your report text and any tables or figures, names of any accompanying files, etc. Remember to commit all the files for your lab submission, to put the URL for your private xv6 repo in the Teams assignment, to submit the Teams assignment, and to give the instructor and TA access to your repo.

Task 1. freepmem() system call

Your modified files to implement the freepmem() system call should be in the hw4 branch of your xv6 repo. Show and explain the results of testing the freepmem() system call using the provided free command and memory-user program. Summarize what you learned by carrying out this task.

Task 2. Change sbrk() so that it does not allocate physical memory.

Commit and push your modified sysproc.c file containing your changed sbrk() system call implementation to your hw4 branch. Show the results of starting xv6 and attempting to run a user command. Explain why the errors occur. Summarize what you learned by carrying out this task.

Task 3. Handle the load and store faults that result from Task 2

Commit and push your modified trap.c file containing your modified usertrap() function to your hw4 branch. Show the results of starting xv6 and running a user command. Explain why errors occur.

Summarize what you learned by carrying out this task.

Describe any difficulties you ran into with this task and if/how you overcame them.

Task 4. Fix kernel panic and any other errors.

List what files you changed to fix the errors and describe the changes. Commit and you’re your modifications to your hw4 branch. Show the results of starting xv6 and running user commands after you have fixed the errors you found.

Summarize what you learned by carrying out this task.

Describe any difficulties you ran into with this task and if/how you overcame them.

Task 5. Test your lazy memory allocation.

Describe the purpose of each of your test cases and show the results. Commit and push your test programs to the user directory in your hw4 branch.

Extra Credit Task 6. Enable use of the entire virtual address space.

List what files you changed and describe the changes.

Summarize what you learned by carrying out this task.

Extra Credit Task 7. Allow a process to turn memory overcommitment on and off.

Describe your approach to this task. Commit and push your code changes to your hw4 branch. Show the results of a test program that turns memory overcommitment on and off.

Summarize what you learned by carrying out this task.