

Assignment

Part I (linux/)

1. Make a new directory called 220 in your home directory
`mkdir 220`
2. Create a symbolic link in your ~/220 called linux to
/home/mgwhite/homescratch/linux
`cd 220`
`ln --symbolic /home/mgwhite/homescratch/linux linux`
3. What are the permissions of the symbolic link in your ~/220?
`ls -l linux`
`lrwxrwxrwx`
Symbolic links have “dummy” permissions. The real permissions are kept with whatever is being pointed to by the symbolic link.
4. What are the permissions of /home/mgwhite/homescratch/linux?
`ls -ld /home/mgwhite/homescratch/linux`
`drwxr-xr-x`
The owner may enter the directory and create, rename, and delete files within the directory. Members of the owner group and world may enter the directory but cannot create, delete, or rename files.
5. Use the command for estimating file space usage to estimate the file space usage of /home/mgwhite/homescratch/linux—what is the total in human readable format?
`du /home/mgwhite/homescratch/linux`
`4772958`
`du -h /home/mgwhite/homescratch/linux`
`4.6G`

Part II (your/subsystem/)

1. Summarize the purpose of your subsystem.
The kernel subsystem contains the generic kernel code that doesn't fit anywhere else such as the scheduler, the `printk()` code, and the signal handling code.
2. What is the size (in human readable format) of the largest .c file (in your subsystem)?
`find /home/mgwhite/homescratch/linux/kernel/ -name '*.c' -type f -print0 | xargs -0 du -h | sort -h`
`201K`

3. What is the size (in human readable format) of the smallest .c file (in your subsystem)?
find /home/mgwhite/homescratch/linux/kernel/ -name '*.c' -type f -print0 | xargs -0 du -h | sort -h -r
4.5K
4. How many lines in .c files (in your subsystem) use the auto keyword?
find /home/mgwhite/homescratch/linux/kernel/ -name '*.c' -type f -print0 | xargs -0 grep -P '(?![\w"]-)]auto(?![\w"]-)]'
0 lines
5. Produce a sorted list of .c files (in your subsystem) that use the typedef keyword.
find /home/mgwhite/homescratch/linux/kernel/ -name '*.c' -type f -print0 | xargs -0 grep -P '(?![\w"]-)]typedef(?![\w"]-)]' | sort > sorted_list_of_c_files.txt
6. Lexically analyze the .c files (in your subsystem) and produce a frequency distribution of lexical elements.
vi rules.l
flex rules.l
gcc lex.yy.c
find /home/mgwhite/homescratch/linux/kernel/ -name '*.c' -type f -print0 | xargs -0 sort | ./a.out | tr ' ' '\n' | grep '\S' | sort | uniq -c | sort -n -r > frequency_distribution.txt
7. What is gcc?
gcc is the GNU project C and C++ compiler. gcc allows for the preprocessing, compilation, assembly, and linking of files.
8. Where is gcc located?
which gcc
/usr/bin/gcc
9. Use gcc to preprocess one of the .c files (in your subsystem) to produce a .i; then compile the .i to produce a .s; then assemble the .s to produce a .o—using the -E, -S, and -c options, respectively.
gcc -I/home/mgwhite/homescratch/linux/include/ -E /home/mgwhite/homescratch/linux/kernel/bounds.c > ~/220/bounds.i
gcc -S bounds.i
gcc -c bounds.s