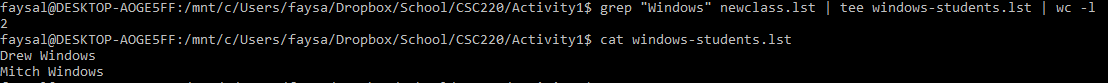
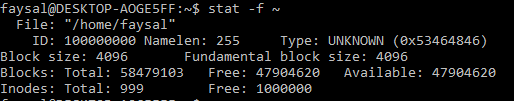
Faysal Khatri

CSC220 -- Activity 2

2017-06-05

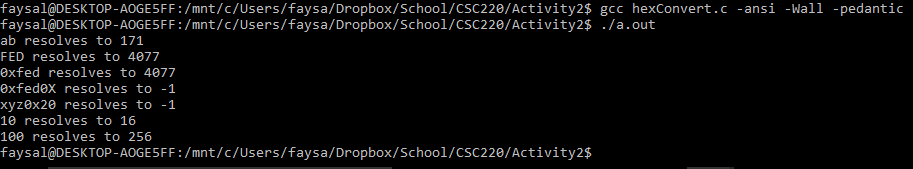
# Part 1: Unix

Question 2

1. sort list.txt > sorted.txt
2. cat > notes.txt (ctrl + d to quit and save)
3. sort list1.txt list2.txt list3.txt -u > alllist.txt
4. grep "Windows" newclass.lst | tee windows-students.lst | wc -l  
   
5. find ~ -name \*.c -print
6. stat -f ~ says the block size is 4kb, so we can’t get resolution down to 1000 bytes.  
     
   find ~ -name \*.c -size +0 -print  
   
7. ls | sort

# Part 2: hex to decimal

## Sample Output



## hexConvert.c

#include <stdio.h>

#include <string.h>

int htoi(char []);

int power16(int);

int main() {

char \*test1 = "ab";

char \*test2 = "FED";

char \*test3 = "0xfed";

char \*test4 = "0xfed0X";

char \*test5 = "xyz0x20";

char \*test6 = "10";

char \*test7 = "100";

printf("%s resolves to %d\n", test1, htoi(test1));

printf("%s resolves to %d\n", test2, htoi(test2));

printf("%s resolves to %d\n", test3, htoi(test3));

printf("%s resolves to %d\n", test4, htoi(test4));

printf("%s resolves to %d\n", test5, htoi(test5));

printf("%s resolves to %d\n", test6, htoi(test6));

printf("%s resolves to %d\n", test7, htoi(test7));

return 0;

}

int htoi(char \*hex) {

int startIndex, endIndex;

int place=0;

int result=0;

int i;

/\*Find out if leading 0x is present\*/

if ( (hex[0] == '0') && ( (hex[1] == 'x') || (hex[1] == 'X') ) ) {

startIndex = 2;

}

else {

startIndex = 0;

}

endIndex = strlen(hex) - 1;

for (i = endIndex; i>=startIndex; i--) {

int digit;

if ( (hex[i] >= 'A' && hex[i] <= 'F') || (hex[i] >= 'a' && hex[i] <= 'f') ) {

switch (hex[i]) {

case 'A': case 'a':

digit = 10;

break;

case 'B': case 'b':

digit = 11;

break;

case 'C': case 'c':

digit = 12;

break;

case 'D': case 'd':

digit = 13;

break;

case 'E': case 'e':

digit = 14;

break;

case 'F': case 'f':

digit = 15;

break;

}

}

else if ( hex[i] >= '0' && hex[i] <= '9') {

digit = (hex[i] - '0');

}

else {

return -1;

}

result += (digit \* power16(place) );

place++;

}

return result;

}

int power16(int a) {

int result = 1;

int j = 0;

for (j=0; j<a; j++) {

result = result \* 16;

}

return result;

}