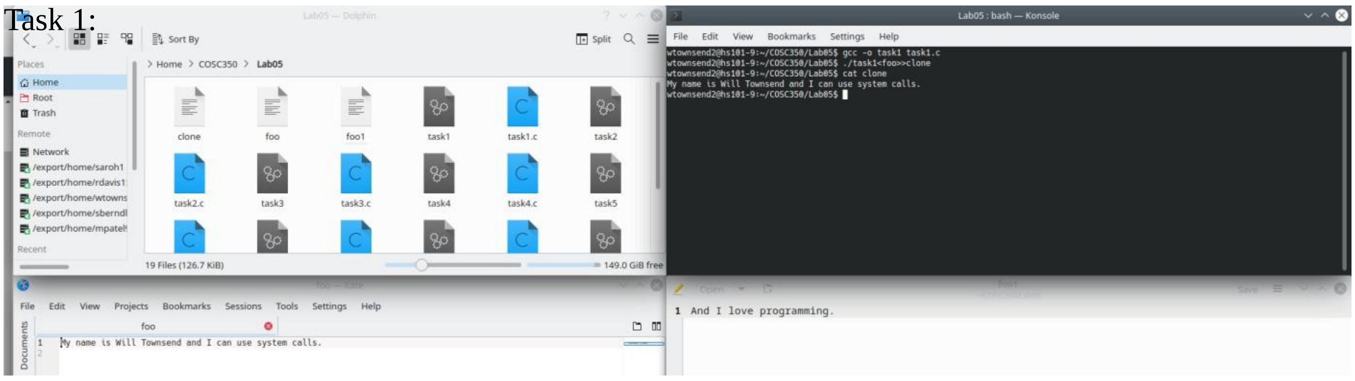
Task 1:



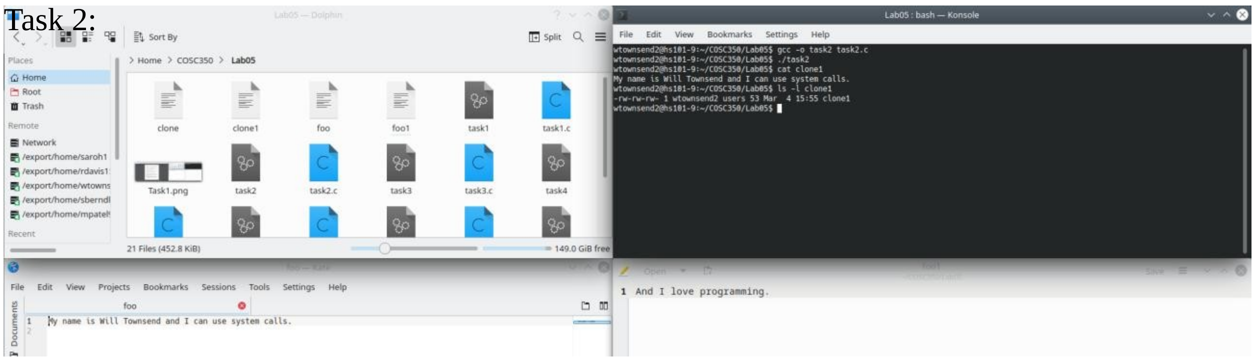
```
wtownsend2@hs101-9:~/COSC350/Lab05$ gcc -o task1 task1.c
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task1<foo>>clone
wtownsend2@hs101-9:~/COSC350/Lab05$ cat clone
My name is Will Townsend and I can use system calls.
wtownsend2@hs101-9:~/COSC350/Lab05$
```

```
My name is Will Townsend and I can use system calls.
```

```
And I love programming.
```

```
/*
* Lab 3 Task 1
* COSC 350
* Will Townsend
*/
#include<stdlib.h>
#include<unistd.h>
#include<fcntl.h>

int main(){
   char buffer[128];//128 byte buffer
   int bytes;//number of bytes read

     //loops through the standard (user) input 128 bytes at a time
   while ((bytes=read(0,buffer,128))>0){
          write(1,buffer,bytes); //writes the bytes to the standard output
     }
     exit(0);
}
```

Task 2:



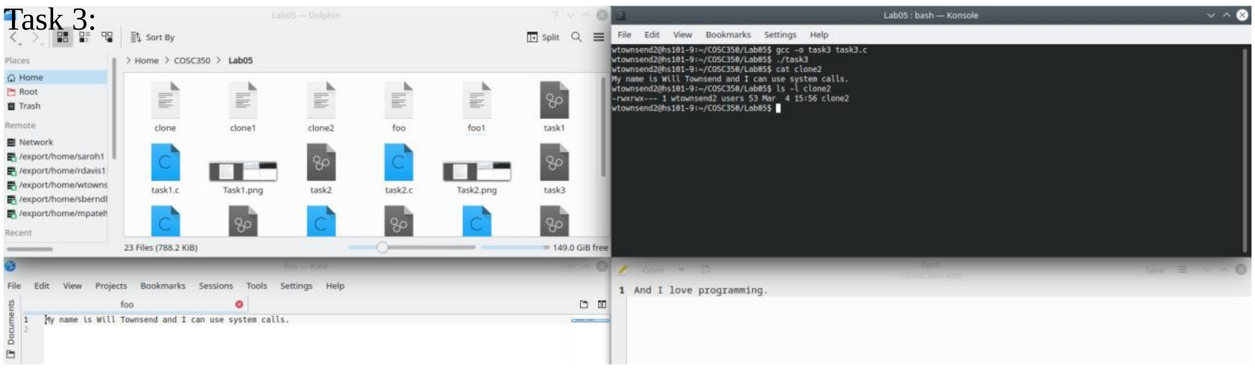
```
wtownsend2@hs101-9:~/COSC350/Lab05$ gcc -o task2 task2.c
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task2
wtownsend2@hs101-9:~/COSC350/Lab05$ cat clone1
My name is Will Townsend and I can use system calls.
wtownsend2@hs101-9:~/COSC350/Lab05$ ls -l clone1
-rw-rw-rw- 1 wtownsend2 users 53 Mar  4 15:55 clone1
wtownsend2@hs101-9:~/COSC350/Lab05$
```

```
1  My name is Will Townsend and I can use system calls.
2
```

```
1  And I love programming.
```

```
/*
* Lab 3 Task 2
* COSC 350
* Will Townsend
*/
#include<stdlib.h>
#include<unistd.h>
#include<fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>

int main(){

    umask(000);//allows all permissions to be activated for usr, grp, or oth
  int fileIn=open("foo",O_RDONLY);//opens a file called foo w/read only permission
  int fileOut=open("clone1", O_CREAT | O_WRONLY,0666);//creates a file clone1 with rw-rw-rw permissions
  char b[1];//one byte buffer
  int rbyte;//number of bytes read

    //reads foo contents one byte at a time and writes them into clone1
  while((rbyte=read(fileIn,b,1)>0)){
        write(fileOut,b,rbyte);
    }

    //closes files opened
    close(fileIn);
    close(fileOut);
    exit(0);
}
```

Task 3:



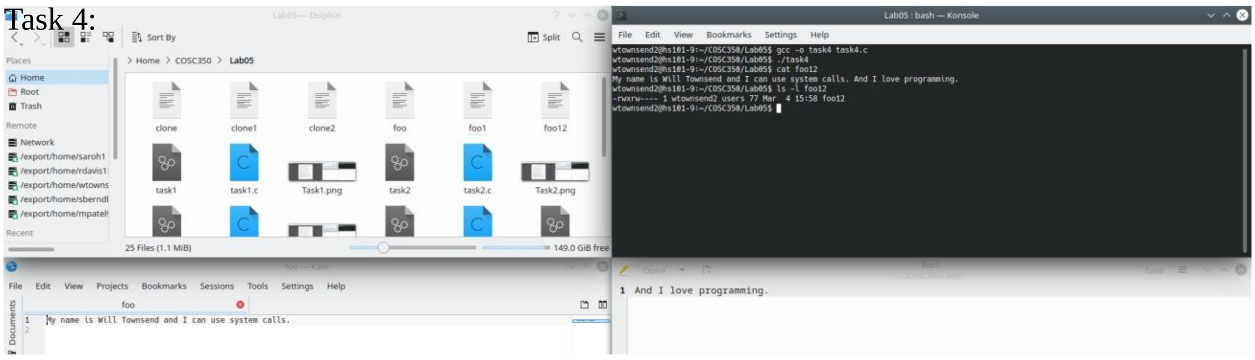
```
wtownsend2@hs101-9:~/COSC350/Lab05$ gcc -o task3 task3.c
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task3
wtownsend2@hs101-9:~/COSC350/Lab05$ cat clone2
My name is Will Townsend and I can use system calls.
wtownsend2@hs101-9:~/COSC350/Lab05$ ls -l clone2
-rwxrwx--- 1 wtownsend2 users 53 Mar  4 15:56 clone2
wtownsend2@hs101-9:~/COSC350/Lab05$
```

```
My name is Will Townsend and I can use system calls.
```

```
And I love programming.
```

```
/*
* Lab 3 Task 3
* COSC 350
* Will Townsend
*/
#include<stdlib.h>
#include<unistd.h>
#include<fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>

int main(){

	umask(000);//allows all permissions to be activated for usr, grp, or oth
    int fileIn=open("foo",O_RDONLY);//opens a file called foo w/read only permission
    int fileOut=open("clone2", O_CREAT | O_WRONLY,0770);//creates a file clone2 with
rwxrwx--- permissions
    char buffer[32];//32 byte buffer
    int rbyte;//number of bytes read

	//reads foo contents 32 bytes at a time and writes them into clone2
    while((rbyte=read(fileIn,buffer,32))>0){
		write(fileOut,buffer,rbyte);
	}

	//closes files opened
	close(fileIn);
	close(fileOut);
	exit(0);
}
```

Task 4:



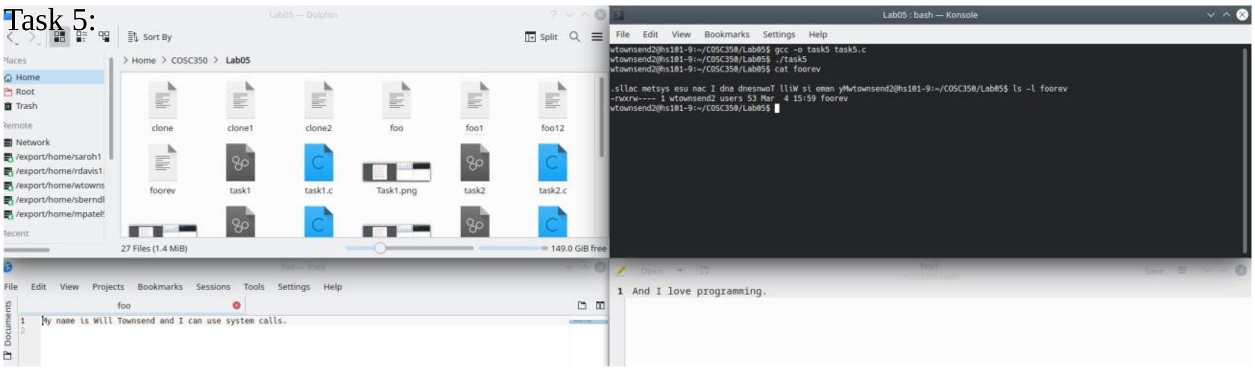
```
wtownsend2@hs101-9:~/COSC350/Lab05$ gcc -o task4 task4.c
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task4
wtownsend2@hs101-9:~/COSC350/Lab05$ cat foo12
My name is Will Townsend and I can use system calls. And I love programming.
wtownsend2@hs101-9:~/COSC350/Lab05$ ls -l foo12
-rwxrw---- 1 wtownsend2 users 77 Mar  4 15:58 foo12
wtownsend2@hs101-9:~/COSC350/Lab05$
```

```
My name is Will Townsend and I can use system calls.
```

```
And I love programming.
```

```
/*
* Lab 3 Task 4
* COSC 350
* Will Townsend
*/
#include<stdlib.h>
#include<unistd.h>
#include<fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>

int main(){

    umask(000);//allows all permissions to be activated for usr, grp, or oth
  int fileIn1=open("foo",O_RDONLY);//opens a file called foo w/read only permission
  int fileIn2=open("foo1",O_RDONLY);//opens a file called foo1 w/read only permission
  int fileOut=open("foo12", O_CREAT | O_RDWR,0760);//creates a file foo12 with rwxrw----
permissions
  char buffer[1024];//1024 byte buffer
  int rbyte;//number of bytes read

    //reads foo contents 1024 bytes at a time and writes them into foo12
  while((rbyte=read(fileIn1,buffer,1024))>0){
        write(fileOut,buffer,rbyte);
    }

    lseek(fileOut,0,SEEK_END);//sets the cursor/offset the end of foo12

    //reads foo1 contents 1024 bytes at a time and writes them into foo12
  while((rbyte=read(fileIn2,buffer,1024))>0){
        write(fileOut,buffer,rbyte);
    }

    //closes files opened
    close(fileIn1);
    close(fileIn2);
    close(fileOut);
    exit(0);
}
```

Task 5:



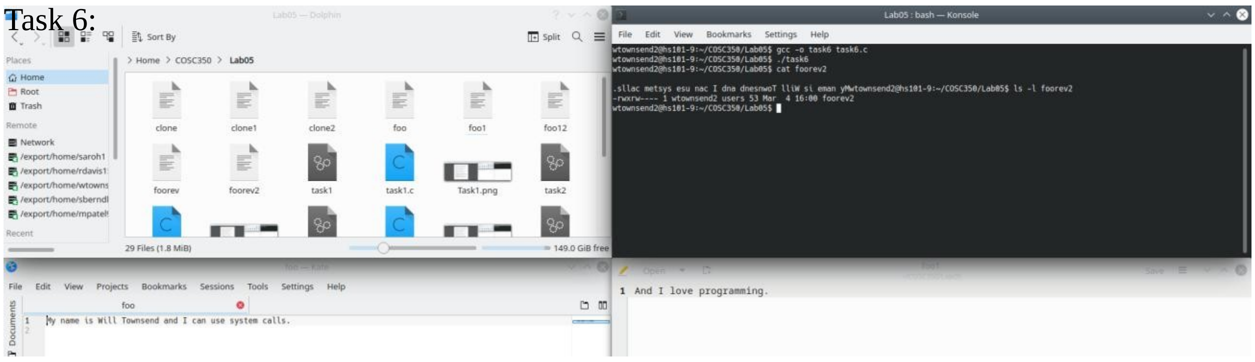
```
wtownsend2@hs101-9:~/COSC350/Lab05$ gcc -o task5 task5.c
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task5
wtownsend2@hs101-9:~/COSC350/Lab05$ cat foorev

.sllac metsys esu nac I dna dnesnwoT lliW si eman yMwtownsend2@hs101-9:~/COSC350/Lab05$ ls -l foorev
-rwxrw---- 1 wtownsend2 users 53 Mar  4 15:59 foorev
wtownsend2@hs101-9:~/COSC350/Lab05$
```

```
My name is Will Townsend and I can use system calls.
```

```
And I love programming.
```

```c
/*
* Lab 3 Task 5
* COSC 350
* Will Townsend
*/
#include<stdlib.h>
#include<unistd.h>
#include<fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>

int main(){

    umask(000);//allows all permissions to be activated for usr, grp, or oth
    int fileIn=open("foo",O_RDONLY);//opens a file called foo w/read only permission
    int fileOut=open("foorev", O_CREAT | O_RDWR, 0760 );//creates a file foorev with rwxrw---- permissions
    char buffer[1];//one byte buffer
    int rbyte;//number of bytes read
    off_t offset=lseek(fileIn,0,SEEK_END)-1;//initial offset (cursor set to the end of the file)

    //reads through foo backwards
    while((rbyte=read(fileIn,buffer,1))>0){
        write(fileOut,buffer,rbyte);//write the current byte
        lseek(fileIn,offset--, SEEK_SET);//moves the cursor back a position
    }

    //closes files opened
    close(fileIn);
    close(fileOut);
    exit(0);
}
```

Task 6:



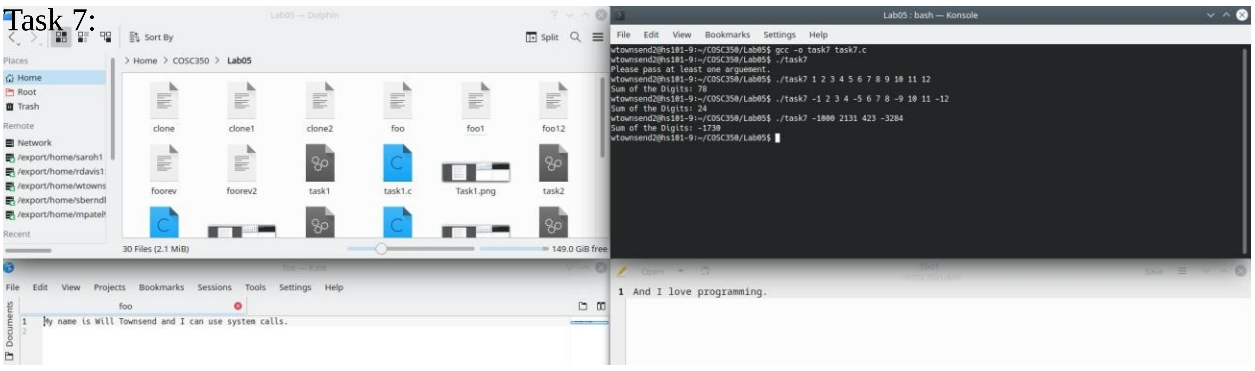
```
wtownsend2@hs101-9:~/COSC350/Lab05$ gcc -o task6 task6.c
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task6
wtownsend2@hs101-9:~/COSC350/Lab05$ cat foorev2

.sllac metsys esu nac I dna dnesnwoT lliW si eman yMwtownsend2@hs101-9:~/COSC350/Lab05$ ls -l foorev2
-rwxrw---- 1 wtownsend2 users 53 Mar  4 16:00 foorev2
wtownsend2@hs101-9:~/COSC350/Lab05$
```

```
My name is Will Townsend and I can use system calls.
```

```
And I love programming.
```

```
/*
* Lab 3 Task 6
* COSC 350
* Will Townsend
*/
#include<stdlib.h>
#include<stdio.h>
#include <sys/types.h>
#include <sys/stat.h>
#include<unistd.h>
#include<fcntl.h>

int main(){
    umask(000);//allows all permissions to be activated for usr, grp, or oth
  int fileIn=open("foo",O_RDONLY);//opens a file called foo w/read only permission
  int fileOut=open("foorev2", O_CREAT | O_RDWR, 0760 );//creates a file foorev2 with
rwxrw---- permissions
  char buffer[1];//one byte buffer
  int rbyte;//number of bytes read
  int i=0;//initializes the file size

    //determines the size of the file (to set the offset correctly)
  while((read(fileIn,buffer,1))>0)
        i++;

    //does the same as task 5 but uses pread to combine the lseek and read functions
  while((rbyte=pread(fileIn,buffer,1,--i))>0)
        write(fileOut,buffer,rbyte);

    //close files opened
    close(fileIn);
    close(fileOut);
    exit(0);
}
```

Task 7:



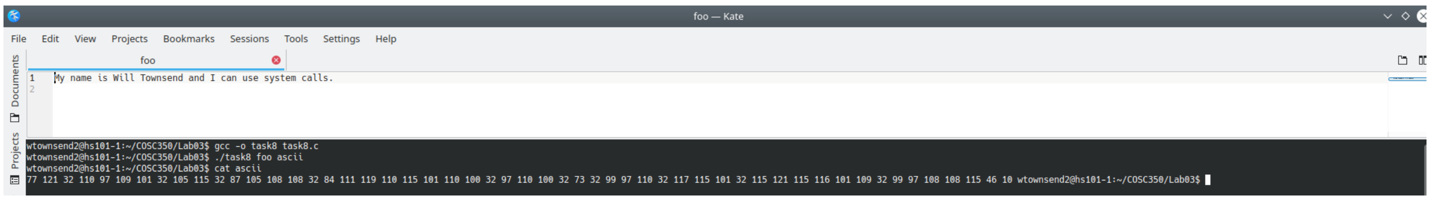
```
wtownsend2@hs101-9:~/COSC350/Lab05$ gcc -o task7 task7.c
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task7
Please pass at least one arguement.
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task7 1 2 3 4 5 6 7 8 9 10 11 12
Sum of the Digits: 78
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task7 -1 2 3 4 -5 6 7 8 -9 10 11 -12
Sum of the Digits: 24
wtownsend2@hs101-9:~/COSC350/Lab05$ ./task7 -1000 2131 423 -3284
Sum of the Digits: -1730
wtownsend2@hs101-9:~/COSC350/Lab05$
```

```
My name is Will Townsend and I can use system calls.
```

```
And I love programming.
```

```
/*
* Lab 3 Task 7
* COSC 350
* Will Townsend
*/
#include<stdlib.h>
#include<stdio.h>
#include<unistd.h>
#include<fcntl.h>

//function to convert a string of digits into an int
int strToInt(char str[]){
   int i=0;//increment variable (for loop)
   int sign=0;//used check sign case
   int num=0;//final number to be returned

     //checks if number is negative, adjusts sign variable accordingly
   if(str[0]=='-')
          sign=-1;
     //adds one to the increment variable to ignore
   if(sign==-1)
          i++;

     //loops through the digits converting each into their digit equivalent
     /*
      * Given a char arr '123' the loops is as follow:
      * 0*10+'1'-'\0'=1 => 1*10+'2'-'\0'=12 => 12*10+'3'-'\0'= 123
      * This loop works for any set of digits in a character array.
     */
   for(i;str[i]!='\0';i++)
          num=num*10 + str[i] - '0';
     //fixes num if '-' was found earlier
   if(sign==-1)
          num=-num;
     //returns the final converted number
   return num;
}

int main(int argc,char *argv[]){
     //error checks the parameters (makes sure there is at leat one integer)
   if(argc<2){
          printf("Please pass at least one arguement.\n");
          exit(1);
     }
   int sum=0;//initializes the sum
     //loops through all the arguements, converts them to integers, and adds them up
   for(int i=1;i<argc;i++)
          sum+=strToInt(argv[i]);
     printf("Sum of the Digits: %d\n", sum);//prints the sum of numbers
     exit(0);
}
```

Task 8:



```
My name is Will Townsend and I can use system calls.
```

```
wtownsend2@hs101-1:~/COSC350/Lab03$ gcc -o task8 task8.c
wtownsend2@hs101-1:~/COSC350/Lab03$ ./task8 foo ascii
wtownsend2@hs101-1:~/COSC350/Lab03$ cat ascii
77 121 32 110 97 109 101 32 105 115 32 87 105 108 108 32 84 111 119 110 115 101 110 100 32 97 110 100 32 73 32 99 97 110 32 117 115 101 32 115 121 115 116 101 109 32 99 97 108 108 115 46 10 wtownsend2@hs101-1:~/COSC350/Lab03$ █
```

```
#include<stdlib.h>
#include<stdio.h>
#include<unistd.h>
#include<fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>

int len(int dig){
    if(dig>=100)
        return 3;
    if(dig>=10)
        return 2;
    return 1;
}

int main(int argc,char *argv[]){
    //error checks the arguements
    if(argc!=3){
        write(2,"Error! Please pass two arguements.\n",36);
        exit(2);
    }

    umask(000);//allows permissions to be set to any for usr, grp, oth
    int fileIn=open(argv[1],O_RDONLY);//opens input file determined by the first arguement
    int fileOut=open(argv[2],O_CREAT | O_RDWR, 0666);//creates the output file (name determined by
second command line arguement

    //error check for files
    if(fileIn==-1 || fileOut==-1){
        write(2,"Error! Cannot open file.\n",26);
        exit(1);
    }

    char b;//singular byte
    int rbyte;//readbytes

    //reading contents of the file
    while((rbyte=read(fileIn,&b,1)>0)){
        int num = (int) b;//conversion of char into an int (ascii form)
        int length=len(num);//grabs the length of the number
        int newB[length];//the buffer containing the digits

        //loops to put all the digits in the buffer
        for(int i=length-1;i>=0;i--){
            newB[i]=num % 10 + '0';//makes it so they are characters
            num=num/10;//moves the digit down by one
        }
        //loops to put the ascii numbers into the file (for some reason I couldnt get it to work with the newB).
        for(int i=0;i<length;i++){
            char b=newB[i];//stores a byte into a smaller buffer
            write(fileOut,&b,1);//writes the digits into the file
        }
        write(fileOut," ",1);//writes a space to seperate each ascii number in the file
    }

    //closes the open files
    close(fileIn);
    close(fileOut);
    exit(0);
}
```