Task 0:

sum.sh


```
#!/bin/bash

sum=0
for i in $(seq 1 $#);do
    eval num=\$$i
    sum=`expr $num + $sum`
done
echo "Sum is $sum"
exit 0
```

fileCount.sh


```
#!/bin/bash

shfiles=`ls -l *.sh | wc -l`
cfiles=`ls -l *.c | wc -l`
cppfiles=`ls -l *.cpp | wc -l`
hfiles=`ls  -l *.h | wc -l`

if [ $shfiles -ne 0 ];then
    echo "There are $shfiles .sh files"
fi

if [ $cfiles -ne 0 ];then
    echo "There are $cfiles .c files"
fi

if [ $cppfiles -ne 0 ];then
    echo "There are $cppfiles .cpp files"
fi

if [ $hfiles -ne 0 ];then
    echo "There are $hfiles .h files"
fi
```
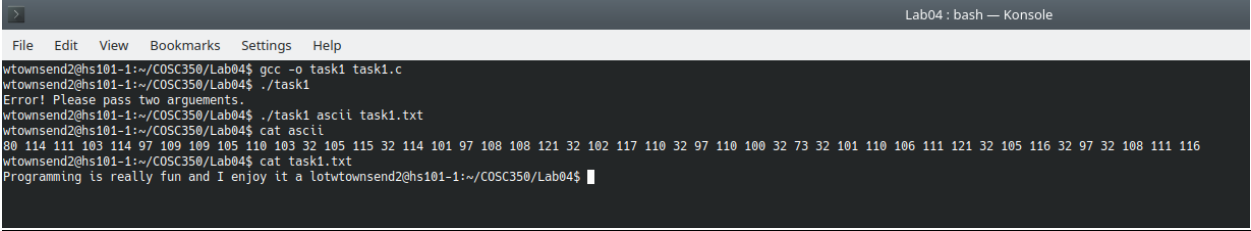
fib.sh


```
#!/bin/bash

num1=0
```

```
num2=1
for i in $(seq 1 $1);do
    echo -n "$num1 "
    newNum2=`expr $num1 + $num2`
    num1=$num2
    num2=$newNum2
done
echo
exit 0
```

Task 1:

File   Edit   View   Bookmarks   Settings   Help

```
wtownsend2@hs101-1:~/COSC350/Lab04$ gcc -o task1 task1.c
wtownsend2@hs101-1:~/COSC350/Lab04$ ./task1
Error! Please pass two arguements.
wtownsend2@hs101-1:~/COSC350/Lab04$ ./task1 ascii task1.txt
wtownsend2@hs101-1:~/COSC350/Lab04$ cat ascii
80 114 111 103 114 97 109 109 105 110 103 32 105 115 32 114 101 97 108 108 121 32 102 117 110 32 97 110 100 32 73 32 101 110 106 111 121 32 105 116 32 97 32 108 111 116
wtownsend2@hs101-1:~/COSC350/Lab04$ cat task1.txt
Programming is really fun and I enjoy it a lotwtownsend2@hs101-1:~/COSC350/Lab04$
```

```c
/*********************
|    Lab 4 Task 1     |
|    Will Townsend    |
|    March  2022      |
*********************/
#include<stdlib.h>
#include<stdio.h>
#include<unistd.h>
#include<fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>

//function converts the arr of digits into a singular int
int intArrToInt(int digits[], int length){
    int finalNum=0;
    for(int i=0;i<length;i++){
        finalNum = finalNum * 10 + digits[i];
    }
    return finalNum;
}

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 buffer
    int rbyte;//read bytes
    int *num;//int array containing the digits
    int index=0;//an index for the num array
    while((rbyte=read(fileIn,&b,1))>0){
        if(b != ' ')//if b is a number
            num[index++]= b - '0';//inserts into the array
        else{
            //this means
            char newB = (char) intArrToInt(num,index);//casts the integer as a
character
            write(fileOut,&newB,1);//writes the character into the file
            index=0;//resets the index for the next set of digits
        }
    }
    //close both of the files that were opened
    close(fileIn);
    close(fileOut);
    exit(0);
}
```
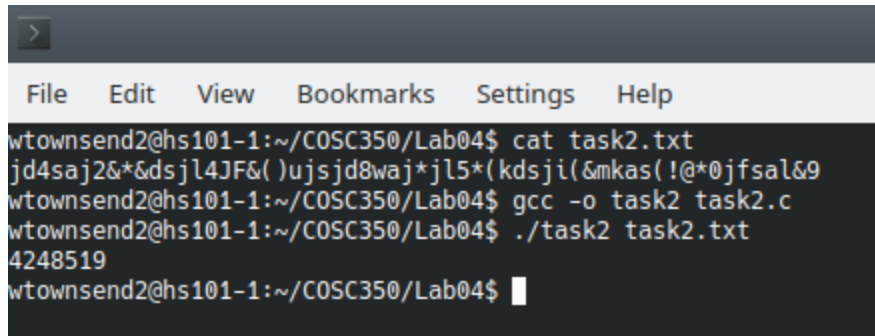
Task 2:


```
/*********************
|    Lab 4 Task 2     |
|    Will Townsend    |
|    March  2022      |
*********************/
#include<stdio.h>
#include<stdlib.h>
#include<fcntl.h>
#include<unistd.h>
#include<ctype.h>
#include<string.h>

/***********************************************
Convert integer to string
Params: x is the int to be converted,
        str is the string into which to write
Returns: length of the string
***********************************************/
int convIntToStr(char * str, int x){
    sprintf(str, "%d", x);
    return (strlen(str));
}

int main(int argc,char *argv[]){
    if (argc!=2){
        puts("Must pass ONE arguement!");
        exit(1);
    }
    int file=open(argv[1],O_RDONLY);
    if (file==-1){
        puts("Error could not open file!");
        exit(2);
    }
    int rbyte;
    char b;
    char *newBuff;
    int sum=0;

    //reads each character in the file to check if they are digits if so convert and
utilize in the variable 'sum'
    while((rbyte=read(file,&b,1))>0)
    if (isdigit((int) b)!=0)
        sum = sum * 10 + b -'0';
    sum+=10;//adds ten to sum
    convIntToStr(newBuff,sum);//calls the function to convert the integer into a
string
    write(1,newBuff,sizeof(newBuff));//writes the embedded integer onto the standard
output
    puts("");
    exit(0);
}
```

Task 3:


```
/*********************
|    Lab 4 Task 3    |
|    Will Townsend   |
|    March  2022     |
*********************/
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<fcntl.h>
#include<sys/stat.h>
#include<sys/types.h>

int palind(int fd1, int fd2){
    char b1;//single byte buffer
    char b2;//another single byte buffer
    off_t offsetFD2=lseek(fd2,0,SEEK_END)-1;//sets the cursor of the file to the last
character
    off_t offsetFD1=0;
    while(offsetFD1!=offsetFD2-1){
        lseek(fd1,offsetFD1++,SEEK_SET);
        read(fd1,&b1,1);
        lseek(fd2,--offsetFD2,SEEK_SET);
        read(fd2,&b2,1);
        if(b1!=b2)
            return 0;
    }

    return 1;//it has looped through successfully and is indeed a palindrome
}

int main(){
    int file=open("task3.txt",O_RDONLY);//opens a test file called task3.txt
    int sameFile=dup(file);//another file descriptor for our opened file
    int res=palind(file,sameFile);//initialized and grabs the result of palind()

    //displays whether the content is a palindrome or not
    if(res==0)
        puts("The file content is not a palindrome!");
    else
        puts("The file content is a palindrome!");
    exit(0);
}
```

Task 4:


```
wtownsend2@hs101-1:~/COSC350/Lab04$ gcc -o task4 hello.c
wtownsend2@hs101-1:~/COSC350/Lab04$ ./task4
wtownsend2@hs101-1:~/COSC350/Lab04$ ls -l ~/Dir2
total 4
drwxr-xr-x 2 wtownsend2 users 4096 Mar 16 14:08 Dir21
wtownsend2@hs101-1:~/COSC350/Lab04$ ls -l ~/Dir2/Dir21
total 4
-rwxr-xr-x 5 wtownsend2 users 32 Mar 16 13:13 hello
wtownsend2@hs101-1:~/COSC350/Lab04$ ls -l ~/Dir1
total 0
lrwxrwxrwx 1 wtownsend2 users 40 Mar 16 14:08 toDir21 -> /mnt/linuxlab/home/wtownsend2/Dir2/Dir21
lrwxrwxrwx 1 wtownsend2 users 46 Mar 16 14:08 toHello -> /mnt/linuxlab/home/wtownsend2/Dir2/Dir21/hello
wtownsend2@hs101-1:~/COSC350/Lab04$
```

```c
/*********************
|    Lab 4 Task 4    |
|    Will Townsend   |
|    March  2022     |
*********************/
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include<fcntl.h>
#include<sys/stat.h>
#include<sys/types.h>

int main(){
    //creates a directory Dir1 in home
    if (mkdir("/mnt/linuxlab/home/wtownsend2/Dir1",0755)==-1)
        puts("Error");
    //creates a directory Dir2 in home
    if (mkdir("/mnt/linuxlab/home/wtownsend2/Dir2",0755)==-1)
        puts("Error");
    //creates a directory Dir21 in Dir2
    if (mkdir("/mnt/linuxlab/home/wtownsend2/Dir2/Dir21",0755)==-1)
        puts("Error");

    //creates a file excutable file hello in Dir21
    link("/mnt/linuxlab/home/wtownsend2/COSC350/Lab04/hello","/mnt/linuxlab/home/wtown
    send2/Dir2/Dir21/hello");

    //creates a symbolic link to Dir21 in Dir1
    symlink("/mnt/linuxlab/home/wtownsend2/Dir2/Dir21","/mnt/linuxlab/home/wtownsend2/
    Dir1/toDir21");

    //creates a symbolic link to hello in Dir1
    symlink("/mnt/linuxlab/home/wtownsend2/Dir2/Dir21/hello","/mnt/linuxlab/home/wtown
    send2/Dir1/toHello");
}
```

Task 5:


```
/********************
|    Lab 4 Task 5    |
|    Will Townsend   |
|     March  2022    |
********************/
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<unistd.h>
#include<fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>

int main(int argc,char *argv[]){
    if(argc!=3){
        puts("Must pass two arguements!");
        exit(1);
    }
    struct stat fileType;
    char oldPath[1024];
    char newPath[1024];
    char *pwd="/mnt/linuxlab/home/wtownsend2/COSC350/Lab04/";
    strcpy(oldPath,pwd);
    strcpy(newPath,argv[2]);

    if(open(argv[1],O_RDONLY)==-1){
        puts("File doesn't exist.");
        exit(2);
    }
    if (stat(newPath,&fileType)==-1){
        int len=strlen(newPath)-1;
        int newNameLen=0;
        char newOld[1024];

        while(newPath[len--]!='/')
            newNameLen++;

        char rev[newNameLen];
        char newName[newNameLen];
        int index=newNameLen;
```

```c
            len=strlen(newPath);
            int lenI=len-1;

            for(int i=0;i<len;i++)
                rev[i]=newPath[lenI--];

            len=index;
            for(int i=0;i<len;i++)
                newName[i]=rev[--index];
            strcat(oldPath,argv[1]);

            rename(oldPath,newPath);
            link(oldPath,newPath);
            unlink(oldPath);
        }
    else{
        if(S_ISDIR(fileType.st_mode)){
            strcat(oldPath,argv[1]);
            strcat(newPath,"/");
            strcat(newPath,argv[1]);
            if (link(oldPath,newPath)==-1){printf("Error\n");}
            if (unlink(oldPath)==-1){printf("Error\n");}
        }
        else if(S_ISREG(fileType.st_mode)){
            char confir='m';
            while(confir!='n' && confir!='y'){
                puts("\nA file with that name exists.");
                printf("Overwrite File? (y/n): ");
                scanf(" %c",&confir);
                if(confir!='n' && confir!='y')
                    puts("Invalid Input!");
                if(confir=='n')
                    exit(3);
            }
            puts("");
            unlink(newPath);
            int len=strlen(newPath)-1;
            int newNameLen=0;
            char newOld[1024];

            while(newPath[len--]!='/')
                newNameLen++;

            char rev[newNameLen];
            char newName[newNameLen];
            int index=newNameLen;
            len=strlen(newPath);
            int lenI=len-1;

            for(int i=0;i<len;i++)
                rev[i]=newPath[lenI--];

            len=index;
            for(int i=0;i<len;i++)
                newName[i]=rev[--index];

            strcat(newOld,oldPath);
            strcat(oldPath,argv[1]);
            strcat(newOld,newName);

            rename(oldPath,newOld);
            link(oldPath,newPath);
            unlink(oldPath);
        }
        else{
            puts("Unknown Error");
            exit(4);
        }
    }
    exit(0);
}
```