CS 2240 Coding StandardsSpring 2011
Dr. Trenary, Peter Holvenstat, Nate Shiff
First Pass

// At the top of all your source files, please include the following information.

```
/*                    //this is the symbol for a multi-line comment
CS 2240
Nate Shiff                                  //name
Tuesday, January 18, 2011          //due date
Assignment 1                                          //assignment specs
```

These specifications should include purpose of program and how it is used.

Read letters from a file whose name is provided on the command line. Print data to a new file with the letters reversed. Also, the new file will have a name equal to the reverse of the original file.

Describe the general approach and functions written to accomplish that.

The program will open source and destination files, reporting errors if they exist, then seek to the end of the source file, and write

```
*/
```

Please comment your code throughly. Comments provide information to the reader that may not be obvious.

```
& recTemp = GetRec ( F ); //send a file pointer,
//                                          receive a struct of type RECORD
```

Please use Stevens-style error checking for all system calls. Note that err_sys( ) resides in error.c

```
#include <stdio.h>
#include "apue.h"   //for Stevens-style error checking
int main(){
        FILE* MYFILE;  //ptr to our stream
        if( NULL == fopen(MYFILE, wb+))
        { err_sys("fopen error");} //truncate  length to 0 to RW

        if(-1==chdir("/home/ugrad/nrshiff")){//some funcs return -1
        //                                                      for error condition.
        //                                                      Refer to Stevens
        //                                                      to find out ;-)
                err_sys("chdir error");
        }
        return 0;       //main is an int function, you know ;-)
}//end main()
```

/ Any function should have its input / output behavior described including variables*

sent and returned, as in

ReverseBuf will reverse, in place, the NumChars characters pointed to by char * Buf, and reverse them in place. No value is returned, no error condition is reported.

```
 */
void ReverseBuf(char * Buf, int NumChars)
            // reverse a buffer's worth or less
{ int i,j; char C;
   j = NumChars -1; i= 0; // subscripting first and last items in buffer
 while (i < j)
  { C = Buf[i]; Buf[i] = Buf[j]; Buf[j] = C; i++; j--; }
}
```

Can any code remain uncommented? Yes. For example, the following program is "self-explanatory." Code of any complexity should include explanatory comments

```
#include <stdio.h>
        #include <math.h>
        int main(){
                double num1=-42;
                double num2=0.123;
                printf("\n");
                printf("num1:%d\nnum2:%d\n",num1,num2);
                printf("sqrt(num1):%d\n",sqrt(num1));
                printf("sqrt(num2):%d\n",sqrt(num2));
                printf("\n");
                return 0;
        }//end main()
```