```c
/*  more01.c  - version 0.1 of more
 *      read and print 24 lines then pause for a few special commands
 */
#include <stdio.h>
#include <stdlib.h>
#define  PAGELEN        24
#define  ERROR          1
#define  SUCCESS        0
#define  has_more_data(x)   (!feof(x))

int  do_more(FILE *);
int  how_much_more();
void print_one_line(FILE *);

int main( int ac , char *av[] )
{
        FILE    *fp;                    /* stream to view with more     */
        int     result = SUCCESS;       /* return status from main      */

        if ( ac == 1 )
                result = do_more( stdin );
        else
                while ( result == SUCCESS && --ac )
                        if ( (fp = fopen( *++av , "r" )) != NULL ){
                                result = do_more( fp ) ;
                                fclose( fp );
                        }
                        else
                                result = ERROR;
        return result;
}

/*  do_more -- show a page of text, then call how_much_more() for instructions
 *      args: FILE * opened to text to display
 *      rets: SUCCESS if ok, ERROR if not
 */
int do_more( FILE *fp )
{
        int     space_left = PAGELEN ;          /* space left on screen */
        int     reply;                          /* user request         */

        while ( has_more_data( fp ) ) {         /* more input   */
                if ( space_left <= 0 ) {                /* screen full? */
                        reply = how_much_more();        /* ask user     */
                        if ( reply == 0 )               /*    n: done   */
                                break;
                        space_left = reply;             /* reset count  */
                }
                print_one_line( fp );
                space_left--;                           /* count it     */
        }
        return SUCCESS;                                 /* EOF => done  */
}

/*  print_one_line(fp) -- copy data from input to stdout until \n or EOF */
void print_one_line( FILE *fp )
{
        int     c;

        while( ( c = getc(fp) ) != EOF && c != '\n' )
                putchar( c ) ;
        putchar('\n');
}

/*  how_much_more -- ask user how much more to show
 *      args: none
 *      rets: number of additional lines to show: 0 => all done
 *      note: space => screenful, 'q' => quit, '\n' => one line
 */
int how_much_more()
{
        int     c;

        printf("\033[7m more? \033[m");         /* reverse on a vt100   */
        while( (c = getchar()) != EOF )
        {
                if ( c == 'q' )                 /* q -> N               */
                        return 0;
                if ( c == ' ' )                 /* ' ' => next page     */
                        return PAGELEN;         /* how many to show     */
                if ( c == '\n' )                /* Enter key => 1 line  */
                        return 1;
        }
        return 0;
}
```