LETTERS

Screen-Oriented Bugs

Dear Marlin,

I want to thank you and all the people at *Dr. Dobb's Journal* for publishing my article, "A Portable, Screen-Oriented Editor" (*DDJ* #63).

Your readers should know about several bugs and typos that have been found in that article. Mr. David Harper of Henrietta, New York gets my nomination as "exterminator of the month" for his work in finding bugs 1 through 8 below. So far, he is the *only* person who has written to tell me about any bugs in the editor.

Bug 1 In the function search1() on file ed3.c the code that reads:

```
if (pat[0] == EOS) {
    return;
}
```

should be changed to:

```
if (pat[0] == EOS) {
        return -1;
}
```

This bug has a small effect on the find and search command-mode commands.

Bug 2 In the function pmtfile() on file ed7.c, change the line:

```
pmtcol1();
```

to:

```
pmtcol1(x);
```

This bug causes the prompt line to contain an erroneous column number following the load or name command-mode commands.

Bugs 3, 4, 5, and 6 I managed to botch the routine syscstat() on file ed8.c. The version of this routine for CP/M 2.2 which was published in *Dr. Dobb's Journal* does work, but it doesn't do what the comment says it should. In addition, the version of syscstat() for CP/M 1.4 contains a separate bug. In the interest of clarity, I won't print the erroneous versions of syscstat() here.

The correct version of syscstat() for CP/M 2.2 is:

```
/* return -1 if no character is ready
   from the console.
 * otherwise, return the character.
 */
syscstat()
{
    int c;
```

Editorial

Electronic Graffiti

Telecommunications is one of the most important uses for microcomputers. It can mean many things: locally linked machines in a separate-but-equal file sharing mode, or in a master-slave(s) relationship; hooking in to centrally controlled data bases; or networking remote micros as independent nodes, over telephone lines or through the airwaves. But it can mean only two things: power to control information, and power accrued by access to information.

Commercial data bases like the Source, MicroNet, and DIALOG are among the telecommunications schemes which get the most press coverage, thanks to big financial interests. Decisions about which information to provide, how easily, at what price (usually between $5 and $100 per access hour), and to which audiences, are effectively made in the corporate boardroom.

Local networks like Nestar, Ethernet and others grow out of "futuristic" office concepts. These are commercial services which link computer workstations under one roof. They allow inter-communication; shared access of files on, for example, hard disk; and monitoring of each operator's actions, especially useful in supervisory or educational environments. Local networks are installed at prices which are low enough, relative to the computing power created, for most financially stable institutions.

Of growing interest, and especially important for computer hobbyists, are the hundreds of no-charge Computer Bulletin Board Systems (CBBS is actually a trademarked acronym). Around the world, microcomputer enthusiasts are putting up CBBS's as a public service, for the interaction with others, and for the experience. They use telephone lines and their own one- and two-disk drive configurations to link users in active interchanges, to provide public notices, articles, and conferences. Some are operated for use by special interest groups in the sciences, computers, arts and politics.

The importance of such homegrown networks should in no way be underestimated. They are quickly mastered, thanks to their ease of operation. Because they are free of charge, there is no stratification of users by economic range. An unwritten operator's code of ethics severely restricts censorship or "editing" of messages, creating a forum for sometimes radical, sometimes ingenious thought.

The foundation is already laid for scenarios from a science author's imagination. At night, humans go to bed as their machines blink quietly, passing messages from one to the next. Disk drives store personal mail and any unprotected through-messages with embedded, pre-selected keywords. A chain of local transmissions travels electronically across several states, avoiding long-distance fees. Broadcast repeaters transmit cross- or inter-continental files, with stringent error checking at each node. CBBS's unite periodically to share selected resources, giving each local user indirect access to an entire world of information.

In our Information Age, those who control information are the ones with real power – personal and political. Given such a scenario, microcomputer-controlled message systems are like local power generators; anyone with a terminal or a micro can use the resultant energy free of charge.

Marlin Ouverson

Editor

```
        c = cpm(6,-1);
        if (c==0) {
                return -1;
        }
        else {
                return c;
        }
}
```

The correct version for syscstat() for CP/M 1.4 is:

```
syscstat()
{
    if (bios(2,0)==0) {
            return -1;
    }
    else {
            return syscin();
    }
}
```

Please note that I still have no way to test the CP/M 1.4 version of syscstat(). Proceed with caution.

Warning And again I say, *warning*. If you make these changes to syscstat(), then you must also change the code in chkkey() on file ed3.c. The old code is:

```
if (c == 0) {
    /* no character at keyboard */
    return(NO);
}
```

The code should be:

```
if (c == -1) {
    /* no character at keyboard */
    return(NO);
}
```

Bug 7 In the function bufdeln() on file ed10.c the code that reads:

```
if (bufdn()==ERR) {
    bufline=oldline;
    oldbufp=bufp;
    return(ERR);
}
```

should be changed to:

```
if (bufdn()==ERR) {
    bufline=oldline;
    bufp=oldbufp;   /* bug fix */
    return(ERR);
}
```

This bug fouls up error recovery.

Bug 8 The following line of bufrepl() should be deleted:

```
bufpmax=bufpmax+n-oldlen;
```

The line already appears in the routine bufext(). This bug makes parts of the main memory buffer unusable as editing progresses. Do not change the bufext() routine!

Bug 9 I did manage to find one bug

Figure 1
Diskdoc Changes

```
/*
 *  get character, no echo
 *  upper case is converted to lower
 */

conlower()
{   char ch;
    ch=conin();
    if ((ch>='A')&(ch<='Z')) ch=ch+'a'-'A';
    return ch;
}


/*
 *  select drive, drive name is 'a','b' etc.
 *  set values for track, sector counts and first sector
 *  return true if ok
 *  to find out what this version does,
 *  refer to the cp/m 2.0 alteration guide
 */

seldrv(drv,pt,ps,pf)
    char drv;
    int *pt,*ps,*pf;        /* where to put track, sector and firstsector */
{   int *dph,*spt,*dsm,*off,halfsecs,trks;
    char *dpb,*bls;
    if ((dph=bios(9,drv-'a'))==0) return 0;
    xlt=dph[0];             /* look at disk parameter header */
    dpb=dph[5];
    if (xlt) *pf=xlt[0];
    else *pf=255&bios(16,0);
    spt=&dpb[0]; *ps=*spt;  /* and at disk parameter block too */
    bls=&dpb[2];
    dsm=&dpb[5];
    off=&dpb[13];
    /* this is tricky since unsigned divide isn't supported */
    halfsecs=(*dsm+1)<<(*bls-1);
    trks=((halfsecs/(*spt))*2)+(((halfsecs%(*spt))+(*spt-1))/(*spt));
    *pt=trks+*off;
    return 1;
}


/*
 *  bios call
 *  will only work with the original small C
 *  other C compilers will usually have the argument sequence reversed
 */

bios(fun,arg)
    int fun,arg;
{   char *ofs;
    ofs=(fun-1)*3;
#asm
    pop     d           ;ofs
    pop     h           ;ret
    pop     b           ;arg
    push    b
    push    h
    push    d
    lhld    1           ;get pointer to bios
    dad     d           ;add offset
    lxi     d,retn1
    push    d
    mov     d,b         ;arg in de too
    mov     e,c
    pchl                ;go
retn1:
    xchg
    mov     l,a
    mvi     h,0
    pop     b           ;ofs
    push    b
    mov     a,c
    cpi     (9-1)*3     ;select disk function?
    jz      retn2
    cpi     (16-1)*3    ;sector translate?
    jnz     retn3
retn2:
    xchg                ;value came in hl
retn3:
#endasm
}
```

on my own. In the routine edjoin() on file ed4.c change the lines:

```
/* abort if the screen isn't wide e-
   nough */
if (k>SCRNW1) {
    return;
}
```

to:

```
/* abort if the screen isn't wide e-
   nough */
if (k>SCRNW1) {
    bufgetln(editbuf, MAXLEN);
    return;
}
```

The effect of this bug is striking when it happens and you should definitely fix it. You can see the effect of the bug as follows:

1. Use the join key to join two lines whose combined length exceeds the width of the screen. Nothing (apparently) will happen.
2. Move the cursor to the middle of the current line.
3. Make some change to the line from edit mode. You will now see that the line you are editing isn't the line that you should be editing.

There are also several typos in the article that may be confusing:

Typo 1 On page 20 of the article, in the paragraph about the split key, it says,

> "Both these keys also switch the editor to insert mode."

This sentence should be deleted. Somehow it was duplicated from a previous paragraph which concerned the insert up and insert down keys.

Typo 2 In Figure 4 of the article, the function of the s command should be changed from:

> move cursor right to H char K

to:

> move cursor right to <char>

Typo 3 The comment for edattop() on file ed4.c should be changed from:

```
/* return true if the current edit
   line is being
 * displayed on the bottom of the
   screen.
 */
```

to:

```
/* return true if the current edit
   line is being
 * displayed on the top of the
   screen.
 */
```

Yours,
Edward K. Ream
1850 Summit Ave.
Madison, WI 53705

Doctoring Diskdoc

Dear Marlin Ouverson,

Since I submitted the Diskdoc listing (*DDJ* #66), two bugs have been found. The first one concerns compatability with other compilers. While Small-C evaluates expressions from left to right, the Kernighan/Ritchie C-language specification leaves the order of evaluation unspecified. The previous version of the *conlower()*-function depended on the order of evaluation, the version shown as Figure 1 does not.

The second bug was caused by an oversight of mine when interpreting the CP/M BIOS disk parameter information. On some double-density formats (mainly the Intellec MDS II), Diskdoc believed that the first sector was number zero, instead of number one. The enclosed revised versions of *seldrv()* and *bios()* will perform correctly.

Yours sincerely,
Egil Kvaleberg
Studpost 111
N-7034 Trondheim-NTH
Norway

Sad About Sinclair

I am an owner of a Sinclair, ZX81 computer and I am dissatisfied with its operation. I experience frequent, unpre-

Figure 2
Small-C Shrinker

```
/*                                      */
/*      Declare local variables         */
/*      (i.e. define for use)           */
/*                                      */
/* works just like "declglb" but modifies machine stack */
/*      and adds symbol table entry with appropriate */
/*      stack offset to find it again                */
declloc(typ)            /* typ is cchar or cint */
        int typ;
        {
        int k, j, newsp; char sname[namesize];
        newsp=Zsp;
        while(1) {
                if(endst())break;
                if(match("*"))
                        j=pointer;
                        else j=variable;
                if (symname(sname)==0)
                        illname();
                if(findloc(sname))
                        multidef(sname);
                if (match("["))
                        {k=needsub();
                        if(k)
                                {j=array;
                                if(typ==cint)k=k+k;}
                        else
                                {j=pointer;
                                k=2;}
                        }
                else
                        if((typ==cchar)
                                &(j!=pointer))
                                k=1;else k=2;
                newsp=newsp-k ;
                addloc(sname, j, typ, newsp);
                if (match(",")==0) break;
                }
        /* change machine stack */
        Zsp=modstk(newsp) ;
        }
```