

50-52,44

## 最新的 CCED4.0/5.0 故障分析集(续)

### 七、CCED 使用打印机方面故障

九、在 CCED 5.0 中采用 2.13 的汉字库 HZK16, HZK24(?=S, F, H, K, T) 打印文稿时, 发现文稿中所有西文字体都是空白, 这是怎么回事? 如何解决?

笔者按照《CCED 5.0 使用手册》中有关说明, 将 2.13 的汉字库放在 \HZK213 的子目录中, 在此子目录中有 CCED 5.0 的汉字库接口模块 CCED213.COM, 在 CCED 5.0 已经将 CCED213.COM 配置好, 但在 CCED 5.0 中文稿中的西文字符还是打印成空白。笔者将 CCED213.COM 拷贝到 \213 在目录中, 却打印出文稿中的西文字符。后面终于明白过来, CCED 5.0 采用 2.13 的汉字库打印文稿时, 若需打印西文字符, 还要 24 点阵 ASCII 码打印驱动程序 ZF24.COM, 解决方法就是将 \213 子目录中的 ZF24.COM 拷贝到 \HZK213 中, 此时, 再打印文稿发现西文字符可正常打印了。

十、在 CCED 5.0 选择 WPS 2.1 的字库 XSDOS.LPH, 则在模拟显示或打印时, 屏幕一片空白, 无汉字显示, 这是何原因? 如何解决?

采用 CCED 5.0 的配置程序 CCEDLT.EXE 重新配置 CCED 5.0 与 WPS2.1 的接口程序 CCEDSLW.COM, 但故障依旧。用 DOS 6.20 的 SCANDISK.EXE 或 NORTON 的 NDD.EXE 检查 C 盘, 却发现 C 盘中的 XSDOS.LPH 文件有问题, 检测完毕后, 发现 XSDOS.LPH 的文件长度由 4M 多变为 60KB, 原来是 XSDOS.LPH 遭到损伤。重新安装 XSDOS.LPH 后, 再在 CCED 5.0 中模拟显示或打印, 汉字出来了, 故障消除。

TP360.63

□ 曹国钧

### 八、CCED 加密原理与解密方法

CCED 4.0/5.0 增加一个加密存盘功能, 但在实际使用中, 有时会忘记了文件的具体密码, 在这种情况下, 如何才能继续使用这种已加了密的文件呢?

经笔者对 CCED 4.0/5.0 的加密文件进行了分析, 发现可以利用 CCED 4.0/5.0 本身加密功能实现其解密, 无需编制程序进行解密。下面就是笔者巧妙地解密 CCED 文件的方法。

CCED 加密文件的结构如下表。从该表中可看出, 在第 22H 字节存放的译码表基值是整个文件破密的关键, 通过求得一个(或二个、三个)关键字符, 再利用 CCED 本身的存盘加密功能, 就能很方便地为 CCED 中密文件破密。

表: CCED 加密文件的结构

偏移量	功能含义
00-61H	存放 CCED 加密文件标志(07 07 57 41 52 4E 20 3A... 65 20 21 30)
22H	存放译码表基值, 该值由使用者输入的密码和 2DH(或 0DH) 相互异或产生
23-25+n	以密文形成存放的密码, 其中的 n 为输入的密码个数
36+n-文件尾	以密文形式存放的正文

1. 利用 PCTOOLS 的 FILE 菜单中的 Edit/View 功能查看 CCED 加密文件的第 22H(38) 字节的内容, 记下该内容, 若第 22H, 23H 字节的内容为 0DH, 0AH, 则取第 22H 字节为 00H;

2. 将该字节与 2DH 异或, 得出的值(假设为 A), 按以下几种情况讨论:

(1)当  $20H < A \leq 60H$  或  $A > 7AH$ , 则该值所代表的 ASCII 字符为关键字符;

(2)当  $A = 20H$ , 关键字符为任意两个相同的字符中加一个空格, 即 AA 等;

(3)当  $A < 20H$ , 将该值和任大于  $40H$  的 ASCII 字符异或得到一个新值, 这个大于  $40H$  的 ASCII 字符和新值代表的 ASCII 字符就是关键字符;

(4)当  $60H < A \leq 7AH$ , 将该值和任大于  $20H$  且小于  $40H$  的 ASCII 字符异或得到了一个新值, 这个大于  $20H$  而小于  $40H$  的 ASCII 字符和新值代表的 ASCII 字符即为关键字符;

2. 利用 PCTOOLS 的 File 菜单中的 Wordp 功能编辑该加密文件, 删除第一句内容, 即加密标志与 2 保密字, 仅留下文件正文;

3. 启动 CCED 4.0/5.0 编辑该文件, 这时文件为密文, 然后用关键字为密码存盘退出;

4. 再利用 PCTOOLS 的 File 菜单中的 Wordp 功能删除第一句内容, 存盘退出 PCTOOLS 工具, 则此时的文件就是已加密后 ASCII 码的文件。

## 九、CCED 5.0 编辑的安全措施

在 CCED 5.0 中编辑文件时, 无意中按下 CTRL+DEL 复合键, 此时, 文件中的所有汉字均变成了“方块”图形, 开始以为是汉字系统显示有问题, 存盘退出 CCED 后, 重新启动汉字系统, 再在 CCED 5.0 中重新配置显示方式, 都不能解决问题, 故障依旧。这是怎么回事?

由于 CCED 5.0 编辑的文件与 WPS 相兼容, 笔者将该文件拿到 WPS 中编辑, 却发现该文件中汉字又出来了, 存盘退出 WPS 后, 再进入 CCED 5.0 为了保护现场编辑的文件内容而设置的安全措施。为了恢复文件中的汉字显示, 可在 CCED 5.0 中用 CTRL+OH (WPS 兼容键盘方式) 或 ALT+M (CCED 兼容键盘方式) 修改“密写级别”为“A-不设置密写级别”即可。

## 十、CCED 计算方面的故障实例

1. 在 CCED 中作减法运算时, 若它们的差值在 -1 到 +1 之间, 为什么不能得到正确的结果? 如何纠正?

在 CCED 中计算  $5 - 5.01$  结果应该为  $-0.01$ , 但在实际计算后发现 CCED 所得到的不是  $-0.01$ , 而是一连串的“\*”。这是 CCED 设计上一个缺陷。解决该问题的方法就是先将减数和被减数都乘以 10 的 X

方幂使减数与被减数都成为整数再除以 10 的 X 方幂, 这样计算出的结果就是正确的 (注意: X 应取为正整数)。例如: 以上面的例子的数值, 在表中的第四列等于第三列与第二列的差, 在这里  $X=2$ , 则计算表达式为:  $C4 = (C3 * 100 - C2 * 100) / 100$ , 此时, 按 CTRL+C, 则计算的结果就是  $-0.01$  了。

2. 在作除法计算时, 若作为除数的数列中存在数字 0 或空格, 则按 CTRL-C 计算后, CCED 计算到数字 0 或空格就会自动停止, 0 或空格以下的数字将得不到运算, 如何解决该问题?

碰到这个问题, 是因为除数为 0 不能计算。可有两个方法解决这个问题, 即

(1) 在定义行块时, 分段定义位避开 0 或空格, 再分段计算, 但此法比较麻烦;

(2) 先确定一个合适的数 X (大小自定, 但不能影响计算精度), 然后, 将 X 与算式中的除数相加, 这样计算时, 即使除数数列中有 0 或空格, 整个数列均会有一个正确的结果。只不过 0 或空格对应的计算结果是一连串的“\*”, 这是正常现象。例如: 表格中的第八列为第一列与第三列之和除以第四列的值, 计算结果取两位小数, 则可这样处理: 取  $X=0.0001$ , 算式是:  $C8 = \text{ROUND}((C1 + C3) / (X4 + X), 2)$ 。

3. 在 CCED 中参与计算的表格是二维开放式的, 则如何在这样的表格中进行表中的数据计算?

一般情况下, CCED 要求参与计算的表格应是封闭的。若是开放式表格, 因第一列无左边界线, 最后一列无右边界线, 因此, 这两列不能参与计算, 此时, 计算的列号应从第二列开始算起, 即第 n 列应定义为  $Cn-1$ , 若开放式表的第一列与最后一列确实需参与计算, 可用 CTRL+D 制表功能先制作为封闭表格, 然后, 将这两列参与计算, 计算完毕后, 可用 CTRL+D 的抹线 (CTRL+光标键) 功能将这两列的外线抹去。

## 十一、CCED 表格处理方面的故障

1. CCED 具有表格线保护和制表方便等极好的表格处理功能, 而 WPS 具有丰富打印控制排版功能, 如何将它们结合起来, 提高制表与排版打印的速度?

可以这样结合 CCED 和 WPS 使用: 在 CCED 中编辑表格后, 再进入 WPS, 用 CTRL+P 和 CTRL+F 选择路径和文件, 调入 CCED 已编辑好的文件, 去掉非法控制符号并重新用 ^B 进行排版, 但需用 WPS 重新设定行间距, 一般情况下, 若 CCED 行间距为 18

(默认值),则 WPS 需设为6;CCED 行间距为16,则 WPS 需设为2或4,这样,WPS 将打印出实线表格来。若又需要修改表格,如表格撑宽或缩小,重新表格中数据编辑等,此时,需先在 WPS 中采用 N 方式存盘或转换(不能使用 D 方式),然后,返回到 CCED 中进行表格编辑工作,对在 WPS 中已输入一些控制符号不要理睬,然后,存盘退出 CCED,再到 WPS 中打印输出。

2. 在 CCED 4.0/5.0 中如何实现表格中的数据横向或纵向的录入?

CCED 4.0/5.0 提供了极强的表格数据处理功能,若用户按 CTRL+F 键一次或四次,就进入了表格锁线状态,此时,可看到 CCED 的状态行有“锁线-ON”的提示信息。在 CCED 中输入数据时,每录入完一个数据,按一次回车键,光标就自动地移到当前列的下一行,从而实现了数据的纵向录入;而在录入完一个数据后,按 TAB 或 SHIFT+TAB 键后,光标就跳到当前列的后一列或前一列,从而实现了数据的横向录入。

3. 在 CCED 4.0/5.0 中如何实现多栏目编辑和编制非正规的表格?

在 CCED 4.0/5.0 中按 SHIFT+F4 重新设置 CCED 5.0 系统,选择该配置菜单的“5-修改某些参数默认值”,则出现一系列提示问答,在“是否排版[Y/N]”后回答 Y,其它按回车选默认值;再按 CTRL+F 和 INS 键。在 CCED 的状态行上可看到“排版-ON 锁线-ON 插入-ON”的提示信息。此时,就进入了多栏目编辑状态。通过画表格线可将显示屏幕分割成若干个各自独立的部分,然后可在各个栏目中录入与编辑不同的内容,在编辑时,栏目可自动向下延伸,并可随时按 F6 横向扩列,按 SHIFT+F6 压缩多余的列;同时,光标移动与删除键及排版命令在这些多栏目内部起作用,真正地实现了表格栏目内的排版。另外,通过采用多栏目的编辑功能及 F6、SHIFT+F6 可快速编制非正规的表格,再复杂的表格,只要灵活运用以上方法都可方便地制成。

4. 在 CCED 4.0/5.0 中大表格录入数据不太方便,如何解决这一问题?

实际上,充分利用 CCED 提供的多窗口功能,并对大表格采用联动录入数据,可解决大表格录入数据不方便的问题,具体做法是这样的:

(1) 将要编辑的大表格调入 CCED 的工作区,并将光标移到屏幕的左上角。

(2) 按两次 CTRL+W,将屏幕分成四个窗口,此时,右边的两个窗口可实现横向联动,右上窗口锁定横向表头;下面两个窗口可实现上下联动,左下窗口锁定纵向表头。

(3) 在右下窗口录入或修改表格数据,此时,横表头和纵表头与表格一起进行同步联动,在表体中录入数据所在的位置一目了然,不但方便,而且可避免出现录入过程中可能产生的错行或错列的情况。

## 十二、CCED 打印分页的技巧

CCED 4.0/5.0 具有自身的分页功能,但其功能不适合表格文件的分页打印。表格文件之所以不分页编辑有利于表间数据整体计算,分页打印说明表格行数较多,打印在一页上难以容纳。怎样运用 CCED 4.0/5.0 编辑命令分页打印出表格文件呢?一个表格文件要分页打印,一般情况下,每页要有相同的表头(每页表格都要具备的某些行,包括主标题,副标题,纵栏标题等)和表尾(表格下面的备注,为便于操作,在此应包括表格底线),而 CCED 分页打印却不能在第一页打印上表尾,第二页打印出上表头。下面以 CCED 4.0 操作为例,就笔者在实际运用 CCED 分页打印表格文件作点讨论。

进入 CCED 4.0 系统,调入编辑好的表格文件,然后,执行以下步骤:

(1) 用 F8 定义表头文字块(包括主标题,副标题,纵栏标题等),光标移到表格内容下面的适当位置(空白行列),用 CTRL+O 复制定义好的表格文字块;

(2) 用 F8 定义表体(表头以下,表尾以上的内容)文字块,每页定义表体文字块行数可随打印页(准备打印的那一页)的大小、字体、字号等因素确定。光标移到复制表头的下一行,用 CTRL+O 复制定义好的打印页的表体文字块。

(3) 用 F8 定义表尾的文字块,光标移到复制表体的下一行,用 CTRL+O 复制表尾;

(4) 光标移动到适当位置,用 CTRL+P 选2进行打印。

(5) 一页打印完毕,重复步骤(2)和(4),整个表格的分页打印完成。

注意:

(1) 字块复制,移动时,光标要移到相应的位置,每页的行数要一致;

(2) 打印控制命令(如字体,字号,行距等)应尽量放在表内设置; (下转第44页)

# D 数据库 ATADASE

```

??' 找不到 &nnnn.fox 文件——&nnnn.prg 文件未生成!'
skip in 2
loop
end!
??' 正在生成 &nnnn.prg 文件...'
zap
!unfox &nnnn.fox>1.txt
appe from 1.txt sdf
!oca for at('unbind',lowe(occ))#0 && 见说明1
!f eof()
?spac(30)+'&nnnn.prg 已生成!'
skip in 2
loop
end!
zap
eras 11111111.
!copy &nnnn.fox 11111111.fox>nul && 避免装订文件与过程文件同名
!unbind 11111111.fox>1.txt && 分解装订过程文件
append from 1.txt sdf
go top
repl all occ with subs(occ,27,at(' ',occ)-27) && 见说明2
go 4
&& 见说明3
?spac(19)+'共有'+|tr|str(recc()-3)+'个过程文件.'
do while !eof()
mmmm=trim(occ) && 过程文件名
?spac(10)+str(recc()-3)+'正在装订过程文件 &mmmm.prg...'
!f file('&mmmm.prg')
?SPAC(10)+'&mmmm.prg 文件被同名的过程文件覆盖,必须运行本程序重新生成!'
end!
!unfox &mmmm.fox>nul && 反伪编译过程文件

```

```

!echo proc &mmmm.>1.txt && 生成一行"PROC 过程文件名"
!copy /a 1.txt+&mmmm.prg 1.txt>nul && 过程文件前加一行
!copy /a 11111111. +1.txt 11111111.>nul && 过程文件合并
eras &mmmm.fox
eras &mmmm.prg
skip
endd
!copy 11111111. &nnnn.prg>nul
?spac(26)+'装订过程文件 &nnnn.prg 已生成!'
skip in 2
eras 11111111.fox
endd
clea all
eras 1.txt
eras 11111111.
eras cccc.dbf
eras cccc.bak
eras occ.bak
retu
** 说明1,当 UNFOX.EXE 反伪编译装订过程文件时,显示行之一有"UNBIND"字符串.
** 说明2,当 UNBIND.EXE 分解伪编译过程文件时,显示行中伪编译过程文件名
** (含后缀.FOX)显示在第27列上.
** 说明3,当 UNBIND.EXE 分解伪编译过程文件时,显示行的前3行与过程文件名无关.
** 不同版本的 UNFOX.EXE 和 UNBIND.EXE 的显示情况如果与以上说明有不同之处,程
** 序要作相应修改.

```

作者地址:福建省三明市列东三明市委组织部  
(365000)

(上接第52页)

(3) 每页打印之前,可对被打印页内容稍加修改;

(4) 事先将编辑好的表格文件进行备份,以防万一。

## 十三、CCED 中的全角字符 转化为半角的技巧

在日常的中文录入过程中,为提高输入速度,用户一般选用五笔字型或拼音状态下的纯中文输入方式,即全角方式,但有时又希望中文文稿中的数字、西文等是半角字符,用手工完成从全角到半角的转换是比较费事的,如何快速地将文稿中的全角转换为半角字符呢?

笔者在使用 CCED 4.0/5.0 的过程中,发现该软件提供的录音与放音功能可满意地解决这个问题。CCED 4.0/5.0 的录音/放音功能是用来做演示之用的,录音是把键盘操作记录在 CCEDKEY.TMP 中,而放音则是把这些操作取出来,代替键盘的操作,由

计算机自动执行。下面以 CCED 4.0 为例实现文稿中的全角到半角的快速转换。

(1) 执行 CCED FILENAME 进入 CCED 4.0 系统,然后,用 F5 查找与替换所要寻找的字符,然后,按 CTRL+Pgup,使光标走到文件头,再替换下一个字符,依次类推,最后存盘退出 CCED;

(2) 将含有全角的文件调入 CCED 4.0 中,即执行 CCED 文件名/L,此时,CCED 将调出文件并按(1)中的顺序处理整个文件,并有音乐提示,处理完成后自动存盘退出。

注意:

(1) 经过以上方法处理的文件每一行可能长短不齐,可在 CCED 中重新排版即可对齐;

(2) CCEDKEY.TMP 记录了(1)中查找与替换方法,应好好地保存,每次重新进行录音操作时,都会使原来的丢失。

(全文完)

作者地址:国家医药管理局重庆医药设计院  
(630042)