

一个一般性的数据录入程序 `scrstr()`。该函数能自动识别来自键盘的上、下、左、右键及退格键,结合调用程序中简单明了的定位方法,就可以实现 C 编程中的全屏幕编辑修改功能。

`scrstr()`函数通过判断键入的上、下、左、右和退格键键代码来识别键盘。这些特殊的键代码对应如下:

↑ ESC [A
↓ ESC [B
→ ESC [C
← ESC [D
退格 0X08

当获得左、右、退格键代码时,做出相应的动作;当获得上移键时退出函数并返回 1;当获得回车键或下移键时退出函数并返回 0。

`scrstr()`函数必须有 `curses` 函数库的支持,因为库函数 `curses` 的初始化屏幕函数 `initscr()`使得 `getchar()`能读取一系列的单个字符,而不需要等待回车键。

为了说明函数 `scrstr()`的具体应用,笔者编制了一个 `try.c` 程序调用 `scrstr()`函数。它判断 `scrstr()`函数的返回值,以使光标移动到相应的录入项。上述程序 `try.c` 和 `scrstr()`已在 XENIX V 系统下编译运行通过,使用效果良好。

```

/* try.c */
#include <stdio.h>
#include <curses.h>
int j,k;
char vol1[9],vol2[9],vol3[9],vol4[9];
main()
{
    short x,y;
    for(j=0;j<9;j++)
        {vol1[j]='\0';vol2[j]='\0';vol3[j]='\0';}
    system("clear");
    x=10;y=10;
    initscr();
    j=1;
    while(j>0)
    {switch(j){
        case 1:k=scrstr(10,10,"aaaa:",vol1,6);
            if(k==1)j=j-2;break;
        case 2:k=scrstr(12,10,"bbbb:",vol2,2);
            if(k==1)j=j-2;break;
        case 3:k=scrstr(14,10,"cccc:",vol3,3);
            if(k==1)j=j-2;break;
        case 4:k=scrstr(16,10,"dddd:",vol4,8);
            if(k==1)j=j-2;break;
        default:j=-1;break;        }j++;}
    endwin();
    printf("\n vol1 = %s \n vol2 = %s \n vol3 = %3 \n vol4 = %s

```

```

\n ",vol1,vol2,vol3,vol4);
    }
    /* scrstr.c */
scrstr(lin,col,pristr,instr,n)
int lin,col,n;
char pristr[],instr[];
{
    int i,j,len;
    char c;
    char vol[81];
    for(i=0,i<81;i++) vol[i]='\0';
    printf("\x1b[%d;%dH",lin,col);
    printf("%s",pristr);
    len=strlen(pristr)+1;
    strcpy(vol,instr);
    printf("%s",vol);
    printf("\x1b[%d;%dH",lin,col+len);
    i=0;
    while((c=getchar())!='\r')
    {
        if(c==0x08)
            if(i<0){i--;
                j=i;
                while(j<n+1){vol[j]=vol[j+1];
                    if(vol[j]!='\0'){printf("");
                        else{printf("%c",vol[j]);}
                    j++;}
                printf("\x1b[%d;%dH",lin,col+len+i);
                continue;}
            else{printf("");
                continue;}
        }
        if(c==0x1b)
            {c=getchar();c=getchar();
            switch(c){
                case 0x41:printf("\x1b[%d;%dH",lin,col+len);
                    printf("\x1b[K");
                    strncpy(instr,vol,n);
                    printf("%s",instr);
                    return(1);
                case 0x42:printf("\x1b[%d;%dH",lin,col+len);
                    printf("\x1b[K");
                    strncpy(instr,vol,n);
                    printf("%s",instr);
                    return(0);
                case 0x43:if(i<n&&vol[j]!='\0'){i++;
                    else {printf("\b");}
                    continue;
                case 0x44:if(i<0){i--;}
                    else {printf("");}
                    continue;
            }
        }
        if(i<n){vol[j]=c;i++;}
        else{printf("\b ");printf("");printf("\b");}
    }
    printf("\x1b[%d;%dH",lin,col+len);
    printf("\x1b[K");
    strcpy(instr,vol,n);
    printf("%s",instr);
    return(0);
}

```