C   (h   ) PDF

: harderr   :
                                : void harderr(int (*fptr) ()).
        : /*This program will trap disk errors and prompt the user for
action. Try running it with no disk in drive A: to invoke its
functions.*/ #include #include #include #define IGNORE 0 #define
RETRY 1 #define ABORT 2 int buf[500]. /*define the error
messages for trapping disk problems*/ static char *err_msg[] = {
"write protect", "unknown unit", "drive not ready", "unknown
command", "data error (CRC)", "bad request", "seek error",
"unknown media type", "sector not found", "printer out of paper",
"write fault", "read fault", "general failure", "reserved", "reserved",
"invalid disk change" }. error_win(char *msg) { int retval.
cputs(msg). /*prompt for user to press a key to abort, retry, ignore*/
while(1) { retval= getch(). if (retval == ' a' || retval == ' A') {
retval = ABORT. break. } if (retval == ' r' || retval == ' R') {
retval = RETRY. break. } if (retval == ' i' || retval == ' I') {
retval = IGNORE. break. } } return(retval). } /*pragma warn -par
reduces warnings which occur due to the non use of the parameters
errval, bp and si to the handler.*/ #pragma warn -par int handler(int
errval,int ax,int bp,int si) { static char msg[80]. unsigned di. int drive.
int errorno. di= _DI. /*if this is not a disk error then it was another
device having trouble*/ if (ax { /* report the error */
error_win("Device error"). /* and return to the program directly

requesting abort */ hardretn(ABORT). } /* otherwise it was a disk error */ drive = ax amp. 0x00FF. /* report which error it was */ sprintf(msg, "Error: %s on drive %c\r\nA)bort, R)etry, I)gnore: ", err_msg[errorno], ' A' drive). /* return to the program via dos interrupt 0x23 with abort, retry, or ignore as input by the user. */ hardresume(error_win(msg)). return ABORT. } #pragma warn par int main(void) { /* install our handler on the hardware problem interrupt */ harderr(handler). clrscr(). printf("Make sure there is no disk in drive A:\n"). printf("Press any key ....\n"). getch(). printf("Trying to access drive A:\n"). printf("fopen returned %p\n",fopen("A:temp.dat", "w")). return 0. } 100Test