```c
/*
 * CLEARDEBUG.C
 *
 * Clear debug in a program image that has it enabled.
 *
 * Tom Allebrandi,    Inland Steel Research and Development
 * East Chicago, IN   Allebrandi@Inland.Com
 */

#include stdio
#include stdlib
#include string
#include "$ihddef.h"

main(
    int  argc,
    char **argv)
{
    char bufferA[512];
    FILE *fileP;
    long *flagsP;

/*
 * The first program argument can be the name of the program file to patch
 */
if (argc < 2)
    {
    fputs("File to patch? ",stdout);
    gets(bufferA);
    }
 else
    strcpy(bufferA,argv[1]);

/*
 * Open the file for update in place
 */
if ((fileP = fopen(bufferA,"rb+")) == NULL)
    {
    perror("Error opening file");
    abort();
    }

/*
 * Read the first block of the file. This should get both the fixed portion
 * of the header and the Image Transfer Vector. It’s possible that it would
 * not get both, but I’ve never seen it.
 */
if (fread(bufferA,sizeof(bufferA),1,fileP) != 1)
    {
    perror("Error reading file");
    abort();
    }

/*
 * Create a pointer to the image header flags
 */
flagsP = (long*) &bufferA[IHD$L_LNKFLAGS];

/*
 * Make sure that the image is enabled for debug
 */
if ((*flagsP & IHD$M_LNKDEBUG) != IHD$M_LNKDEBUG)
    {
    fprintf(stderr,"Image not enabled for debug!\n");
    abort();
    }

/*
 * Clear the debug bit
 */
*flagsP &= ~IHD$M_LNKDEBUG;

/*
 * Update the image file and we are done!
 */
if (fseek(fileP,0L,SEEK_SET) != 0)
    {
    perror("Error rewinding file");
    abort();
    }

if (fwrite(bufferA,sizeof(bufferA),1,fileP) != 1)
    {
    perror("Error writing file");
    abort();
    }

fclose(fileP);
exit(1);
}
```