

### Reducing program size

In some cases you are more interested in reducing the length of a program rather than increasing its speed. For example, the program may be too long for the available memory or you may want to get more programs on a disk. In this case, don't insert any FAST commands into the program. Typically the Optimizing Compiler will reduce program length by about 20 to 25 percent. In addition, you might consider using the /J option.

### SPECIAL NOTES

(1) While the program is executing FAST sections of code, you normally cannot use the BREAK key to stop the program. If you do press the BREAK key in a FAST section, the system will retain this and execute the break when the program leaves the FAST section.

(2) In FAST sections array subscripts and the values used in ON GOTO and ON GOSUB are not checked to verify that they are within bounds. This increases program speed, but it means that you can't use a TRAP command to check for these errors. If you need to check these bounds, you must do this as part of the program.

(3) If you get a runtime error while executing compiled code, the system will ask you to insert a disk with the program and then enter the program name. Normally the system then gives the error and the line number where the error occurred. For programs compiled with the Optimizing Compiler, the system still gives the error but the line number won't be correct because the program length has been changed.

### Reporting Problems or Errors

If you should encounter a problem or error in the Optimizing Compiler or the manual, we would appreciate hearing about it. Please list the computer you were using and, if possible, a short example of a program which malfunctions. Send to:

JOHN J. COOPER  
2200 E. 15TH AVE  
DENVER, CO 80202

Advan Language Designs  
P.O. Box 159  
Baldwin, Kansas 66006