

1997 ACM Mid-Central Regional Programming Contest

Notes to Teams

1. All output will be judged using a file comparison utility, so output must be *exactly* as shown in the examples. This applies to all problems, whether explicitly stated in the problem description or not. Spelling, punctuation, spacing, and case (uppercase/lowercase) are all significant.
2. The judges will ignore all output to the screen. Only the contents of the specified output file will be judged. You can write as much debugging information to the screen as you want.
3. Your program cannot require any intervention by the user. For example, you cannot pause the program and ask the user to press a key to continue. If you do, you will be flagged with a *Submission Error*.
4. Do *not* use drive and/or path specifications when naming input and output files. If a problem indicates that the input file is named 'file.in', then you must open 'file.in' and not 'a:file.in' or 'c:\stuff\file.in' or anything else. Failure to adhere to this rule will result in a *Submission Error*.
5. All test cases used in judging will conform to the input specifications. It is not necessary for you to detect invalid input.
6. Input files and correct output files will obey the following rules.
 - Other than end-of-line characters, spaces are the only whitespace that appear.
 - There are never two or more consecutive spaces in a line.
 - Spaces do not appear at the beginning or end of lines.
 - Blank lines do not appear.
 - All lines, including the last line in the file, end with an end-of-line marker (in MS-DOS this is the standard carriage-return/linefeed sequence).
7. (This applies only to C/C++ programmers.) If you are using an unfamiliar compiler, be sure that you know the sizes of all primitive data types. For example, in most MS-DOS and Windows 3.1x compilers an `int` is 16 bits, but in Windows 95/NT and Unix compilers an `int` is 32 bits.