("Garbage in, garbage out" is not a law of nature, but a commentary on how well principles (1) and (2) are followed in practice.)

(3) Read input until end-of-file or marker, not by count.

(4) Identify input errors and recover if possible. Do not stop on the first error. Do not simply ignore errors.

(5) Use mnemonic input and output. Make input easy to prepare (and easy to prepare correctly). Echo the input and any defaults onto the output; make output self-explanatory.

(6) Localize I/O instead of spreading it all over the program. Hide the details of end of file, buffering, etc., in functions.

(7) Make sure that program structure reflects the data the program processes.