To Err is Human – A Guide to Managing Computer Errors on Exams

The word **error** entails different meanings and usages relative to how it is conceptually applied. The concrete meaning of the Latin word "error" is "wandering" or "straying". Unlike an illusion, an error or a mistake can sometimes be dispelled through knowledge (knowing that one is looking at a mirage and not at real water does not make the mirage disappear). For example, a person who uses too much of an ingredient in a recipe and has a failed product can learn the right amount to use and avoid repeating the mistake. However, some errors can occur even when individuals have the required knowledge to perform a task correctly. Examples include forgetting to collect change after buying chocolate from a vending machine, forgetting the original document after making photocopies, and forgetting to turn the gas off after cooking a meal. Some errors occur when an individual is distracted by something else. (From Wikipedia, the free encyclopedia)

|  |  |  |
| --- | --- | --- |
| Noun | 1. | programming error - error resulting from bad code in some program involved in producing the erroneous result[software error](http://www.thefreedictionary.com/software%2Berror)[computer science](http://www.thefreedictionary.com/computer%2Bscience), [computing](http://www.thefreedictionary.com/computing) - the branch of engineering science that studies (with the aid of computers) computable processes and structures[computer error](http://www.thefreedictionary.com/computer%2Berror), [error](http://www.thefreedictionary.com/error) - (computer science) the occurrence of an incorrect result produced by a computer[run-time error](http://www.thefreedictionary.com/run-time%2Berror), [runtime error](http://www.thefreedictionary.com/runtime%2Berror), [semantic error](http://www.thefreedictionary.com/semantic%2Berror) - an error in logic or arithmetic that must be detected at run time[syntax error](http://www.thefreedictionary.com/syntax%2Berror) - an error of language resulting from code that does not conform to the syntax of the programming language; "syntax errors can be recognized at compilation time"; "a common syntax error is to omit a parenthesis" |

**In the context of CHEN3600 exams, students are instructed “not to leave a spreadsheet, function, subprogram (macro), or workbook” in error.**

If the spreadsheet is in error, one of the following #NA, #REF, #VALUE, #DIV/0, etc.

If a function or subprogram is in error when not being run, when compiled it identifies a syntactical errors (with appropriate error message). Errors in functions and subprograms can cause the spreadsheet errors above.

If a function or subprogram is in error when being run, there were no compile errors and we have instead a runtime error. Runtime errors occur when impossible situations occur such as reading from a file that does not exist, writing to columns and rows that don’t exist, generation of a number that cannot be stored, violation of math (“cat” + 4 = ???), etc. Runtime errors usually cause a dialog box to be opened with information about the error detected.

Logic errors refer to those errors that cause the program to not perform as expected. The only logic error of concern (in this discussion) are programs that “crash”, “get stuck in endless loops”, or that “interfere with the normal operation of the computer”.

Back to CHEN3600. When a spreadsheet is submitted for grading that contains errors (any of the sort listed above), it becomes difficult or impossible for the instructor to determine what “credit” can be awarded for the work supplied since the state of the solution is indeterminate.

One approach would be to award a problem grade of “0” for any program that has any error (since obviously the program is not doing what is required). However, in the situation that most students would prefer (partial credit) the instructor has to have evidence of the progress made on the solution in order to provide appropriate credit.

Therefore, when an error condition arises, it needs to be immediately “corrected”. The easiest way to do this, and the way suggested many times by the instructor, is “stepwise improvement” where the student maintains “control” of the program from the very beginning and moves the program in the direction of the solution in a methodical, step-by-step improvement fashion. If control is maintained and an error arises, one only needs to (1) consider the recent improvement and identify the reason the error was caused, or (2) undo the most recent improvement and reconsider the next stop to take. THIS ELIMINATES THE ERROR CONDITION. Your program may “still not work”, but it can be evaluated as to its operation.

Final words…

In severe cases, a program with errors such as “getting stuck in a loop” jeopardize the instructor’s grading of the course, in that Excel is being used to record grades.

In order to avoid needless loss of credit just assure that your program stays under your control and errors (no matter what type) are eliminated before proceeding. If you try and try and try to get rid of an error (not the same as trying to get code to work correctly), they you obviously have “lost control” of the program. This is a very undesirable (and very avoidable) situation.