# Parsing GROL Element Question Pool Documents

First, you’ll need to convert/export from the PDF format of the question pool documents. There are all sorts of ways to do that so I won’t spend time on it. I usually export to MS Word.

Next, you’ll need to export to a .txt file so the program can read it. Word and other programs all have the ability to Save As, to text. Once you’ve got it in txt format you’ll want to carve off any beginning or ending text in the documents that is directly related to the exam. Keep the parts that tell you what sub topic it is, etc. We’ll pull those into a table so the user can later choose what topics to test on.

Make sure that the document text is consistently punctuated. In Element 7 I found that the colon character used as a separator between parts of the headers had mistakenly been printed as a period:

[error text]

*Section-A: General Information and System Overview. Key Topic #1: Fundamental Concepts:*

[corrected text]

*Section-A: General Information and System Overview: Key Topic #1: Fundamental Concepts:*

There are various typos in the source text. The Parser will toss an error if it can determine it is just a malformed question and the messagebox will tell you the question number. Just visually inspect the question and the error will usually pretty obvious such as a missing “-“ separator in the question number: 13A3 vs 1-3A3, etc.

Common formatting errors include:

Comma instead of period – A, some answer vs A. some answer

Missing space character – 12B4- A vs 12B4 – A

Manual (forced) carriage return in middle of answer text (only terminating CR is okay)

There’s one sneaky one. It is a space character visually in most text readers but is actually a special character that the program interprets as a “-“ character instead of a space. I have created an error log writer to dump the question makeup to the point of an error. It is located in the directory where the program runs from and called “ErrorLog.txt”. Check the entry there for odd errors.

Keep the cleaned up txt files in case you need to reparse them later for whatever reason. I’ve included the latest versions in the program folder (as of the last build).

# # #