Programming Assignment #1
Command Parser

February 19, 2017

1 Objective

- To develop a program that reads user commands from the command prompt and react upon these commands.

2 Submission Instruction

- You are expected to submit using the online submission system using the upload file(s) link.
- Instructions on how to test your program against the check tool hosted on a local server will be provided shortly.
- Note that the server is **locally hosted and can only be accessed when one is connected to the Facult of Engineering, Cairo University Local Area Network**.

The submitted code file should be named A1.cpp
if your file has a different name, it will not be considered in the evaluation

Submission Deadline is March 2 @ 10PM [Firm deadline].

- Missing the deadline == No Marks for this assignment (No late submission)
- Submit even if your code is partially working
- Copying other people code even from previous years is forbidden and is checked by an automated system. Confirmed cases take a grade of -10 (minus 10) irrespective of who is the original code owner.
- Thinking with others is allowed and encouraged. Copying is what is not allowed.

3 Detailed Operation

- In all the following, a1.exe is assumed to be the name of your executable file

- Typically in the command prompt we write
  a1.exe command [optional parameter(s)]

- Your task is to **identify** the command and its parameters if applicable then **return** the correct output or the proper error message.

- A List of the commands and their expected output is shown below.

3.1 print command

- This command should print all the passed arguments separated by single spaces (indpendent of number of spaces/tabs between them).

- **Command Line:** a1.exe print third   year   students are honest   and   clever
  Expected output: third year students are honest and clever

3.2 reverse command

- This command should print the provided arguments in a reverse order separated by single spaces

- **Command Line:** a1.exe reverse Cats love rabbits but rabbits hate cats
  Expected output: cats hate rabbits but rabbits love cats

3.3 upper command

- This command should print all the provided arguments with all characters lowercase

- **Command Line:** a1.exe upper cairo university
  Expected output: CAIRO UNIVERSITY

3.4 shuffleWord command

- This command should shuffle the characters of a provided word by swapping every pair of characters

- **Command Line:** a1.exe shuffle structure
  Expected output: tsurtcrue

3.5 shuffleStatement

- This command should shuffle the words of the provided statements

- **Command Line:** a1.exe shuffleStatement Egypt has goodies
  Expected output: has Egypt goodies

3.6 Delete

- This command should delete one of the provided arguments and print the remaining arguments. The argument to be deleted is identified by the first parameter given after delete

- **Command Line:** a1.exe delete 3 12 5 good 3 nice
  Expected output: 12 5 3 nice

- Note that 3 represent an index for the third element in the list and the list starts at 12.

- Note also the list may contain any data type

3.7 Middle

- This command finds and prints the middle element(s) in a list of arguments. If the number is arguments after the command is even, then there are two middles.
- **Command Line:** a1.exe middle 3 12 5 good 3 nice
  Expected output: 5 good

3.8 Add

- This command should add all the integer numbers after add and return the sum
- If the arguments are non-numeric integers then it should print an error message as described below in section 3.10.
- **Command Line:** a1.exe add 4 5 6 2
  Expected output: 17

3.9 Random

- Thiscommand should generate a sequence of random numbers give a specific range (min and max numbers) and a seed value. The length of the sequence is passed as a parameter
- **Command Line:** a1.exe random 10 1 10 4
  a1.exe random sequenceLength minNumber maxNumber seedValue

  Expected output: 8 2 10 3 10 1 7 8 7 1
- Use rand() and srand(...) functions in this part

3.10 ERROR Handling

- You should check for the correctness of every command (e.g. number of arguments).
- You may use the following error messages to reflect the reason of not performing the command
  - "Undefined Command"
  - "Incorrect Number of Arguments"
  - "Incorrect Data Type"
- **Note that error messages are case sensitive.**

4 Additonal Information

For a quick tutorial on parsing command arguments have a look at this link or this link