**Sentence Analyzer**

**DOS Prompts**

**Output:** “SENTENCE ANALYZER APP PART 2”.

**Output:** Add a blank line of text.

**IF** sub-directory C:\Text Analysis\Sentence Output Files\ **NOT** created yet **THEN**

Create sub-directory, C:\Text Analysis\Sentence Output Files\.

**Output:** “Default output file folder path: C:\Text Analysis\Sentence Output Files\”.

**Output:** Add a blank line of text.

**ENDIF**

Create the following 2 data files: Sentence-Analysis-DataSet1.txt, Sentence-Analysis-DataSet2.txt.

Store above 2 data files in: C:\Text Analysis\Sentence Output Files\.

Create a file called Complete-Estimated-Text-Sentence.txt and store it in subdirectory:

“C:\Text Analysis\Sentence Output Files\”.

**Output:** “Enter number from 1 to 1000 that represents the number of sentences that you want analyzed then press enter key”.

**Action:** User enters number of sentences they want analyzed then presses enter key.

**Total\_Num\_Sentences** = Number of sentences a user wants analyzed.

**Output:** Add a blank line of text.

**Action:** **GOTO** Sentence Analysis SUBROUTINE.

**Output:** “Analysis started for: C:\Text Analysis\Word Input Files\coca-samples-text.txt”.

**Output:** Add a blank line of text.

**Output:** “Analysis result saved to: C:\Text Analysis\Sentence Output Files\Final-Sentence-Analysis.txt”.



**Output:** Add a blank line of text.

**Output:** “Analysis completed!”.

**Sentence Analysis SUBROUTINE**

1) Randomly pick a given sentence in coca-samples-text.txt (located in C:\Text Analysis\Word Input Files\).

**Total\_Num\_Words\_RCS** = Calculate total number of words in randomly chosen sentence.

**IF** **Total\_Num\_Words\_RCS** < 5 **THEN** **GOTO** **STEP** 1

**Total\_Num\_Words\_Analyzed** = **Total\_Num\_Words\_Analyzed** + (**Total\_Num\_Words\_RCS** – 4)

Add one to value of **Num\_Sentences\_Analyzed** - number of sentences analyzed so far.

**IF** **Num\_Sentences\_Analyzed** > **Total\_Num\_Sentences** **THEN**

Create Final-Sentence-Analysis.txt.

Add Final-Sentence-Analysis.txt to: C:\Text Analysis\Sentence Output Files\.

**GOTO** Final Sentence Analysis SUBROUTINE.

**ENDIF**

**FOR** **Index** = 1 **TO** **Total\_Num\_Words\_RCS** **STEP** 1

**Complete\_Text\_Sentence(Index)** = **Index** word in randomly chosen sentence.

**~~Word\_Length~~** ~~= Calculate length (in characters) of~~ **~~Complete\_Text\_Sentence(Index)~~**~~.~~

**~~Total\_Num\_CTS\_Chars~~** ~~=~~ **~~Total\_Num\_CTS\_Chars~~** ~~+~~ **~~Word\_Length.~~**

**NEXT** **Index**

~~2) Add one to~~ **~~NPAW~~** ~~- indicates number of present analyzed word.~~

**~~Word~~** ~~=~~ **~~Complete\_Text\_Sentence(NPAW)~~**~~.~~

**~~Word\_Length~~** ~~= Determine length (in characters) of~~ **~~Word~~**~~.~~

**~~IF~~****~~Word\_Length~~** ~~= 1~~ **~~THEN~~**

~~Add 1 to~~ **~~Total\_Num\_Written\_Chars~~** ~~- counts total number of characters written by user.~~

**~~First\_Four\_Words\_Sentence(NPAW)~~** ~~= (~~**~~Word~~**~~,”True”).~~

**~~Remark:~~** ~~Whole Word (“True”) – Partial Word (“False”).~~

**~~ENDIF~~**

**~~IF~~****~~Word\_Length~~** ~~= 2~~ **~~THEN~~**

~~Add 2.5 to~~ **~~Total\_Num\_Written\_Chars~~**~~.~~

**~~First\_ Four\_Words\_Sentence(NPAW)~~** ~~= (~~**~~Word~~**~~,”True”).~~

**~~ENDIF~~**

**~~IF~~****~~Word\_Length~~** ~~= 3~~ **~~THEN~~**

~~Add 3.5 to~~ **~~Total\_Num\_Written\_Chars~~**~~.~~

**~~First\_ Four\_Words\_Sentence(NPAW)~~** ~~= (~~**~~Word~~**~~,”True”).~~

**~~ENDIF~~**

**~~IF~~****~~Word\_Length~~** ~~> 3~~ **~~THEN~~**

~~Add 4 to~~ **~~Total\_Num\_Written\_Chars~~**~~.~~

**~~Word~~** ~~= calculate first 4 characters of~~ **~~Word~~**~~.~~

**~~First\_ Four\_Words\_Sentence(NPAW)~~**~~= (~~**~~Word~~**~~,”False”).~~

**~~ENDIF~~**

**~~IF~~****~~NPAW~~** ~~< 4~~ **~~THEN GOTO~~****~~Step~~** ~~2~~

**~~GOTO~~** ~~Matching Phrase Analysis SUBROUTINE.~~

~~Add partial word list entry (along with all related sublist entries) and two blank lines of text to Sentence-Analysis-DataSet1.txt.~~

**FOR** **Index** = 1 **TO** 3 **STEP** 1

Add one to **NPAW**.

**Estimated\_Text\_Sentence(Index)** = **Complete\_Text\_Sentence(Index)**.

**NEXT Index**

**~~IF~~****~~Total\_Num\_Words\_RCS~~** ~~= 4~~ **~~THEN~~**

**~~GOTO~~** ~~Complete Estimated Text Sentence SUBROUTINE.~~

**~~ELSE~~**

**GOTO** Estimated Text Sentence SUBROUTINE.

**Estimated Text Sentence SUBROUTINE**

Add one to **NPAW**.

**IF** **NPAW** = **Total\_Num\_Words\_RCS** **THEN**

**NPAW** = 3.

**GOTO** Phrase Analysis SUBROUTINE

**ENDIF**

**Word** = **Complete\_Text\_Sentence(NPAW)**.

**Word\_Length** = calculate length (in characters) of **Word**.

**Total\_Num\_CTS\_Chars** = **Total\_Num\_CTS\_Chars** + **Word\_Length.**

**IF** **Word\_Length** > 1 **THEN**

Add 2 to **Total\_Num\_Inputted\_Chars.**

**Partial\_Word** **=** Determine text value of first 2 characters of **Word**.

**Estimated\_Text\_Sentence(NPAW)** = **Partial\_Word**.

**ELSE**

Add 1 to **Total\_Num\_Inputted\_Chars.**

**Estimated\_Text\_Sentence(NPAW)** = **Word**.

**ENDIF**

**GOTO** Estimated Text Sentence SUBROUTINE.

**Phrase Analysis SUBROUTINE**

Add one to **NPAW**.

**IF** **NPAW** = **Total\_Num\_Words\_RCS** **THEN**

**~~GOTO~~** ~~Final Phrase Analysis SUBROUTINE~~

**Estimated\_Text\_Sentence(NPAW)** = **Complete\_Text\_Sentence(NPAW)**.

**GOTO** Complete Estimated Text Sentence SUBROUTINE.

**ENDIF**

**Word** = **Completed\_Text\_Sentence(NPAW)**.

**Word\_Length** = calculate length (in characters) of **Word**.

**IF** **Word\_Length** = 1 **THEN**

**GOTO** Phrase Analysis SUBROUTINE

**ENDIF**

**WW1** = **Estimated\_Text\_Sentence**(**NPAW** – 3).

**WW2** = **Estimated\_Text\_Sentence**(**NPAW** – 2).

**WW3** = **Estimated\_Text\_Sentence**(**NPAW** – 1).

**PW4** = **Estimated\_Text\_Sentence**(**NPAW**).

**Partial\_Word\_List\_Entry** = **WW1** + **spc** + **WW2** + **spc** + **WW3** + **spc** + **PW4**.

**Remark:** **spc** = one space character.

**Total\_Num\_Partial\_Characters** = 2.

**GOTO** Matching Phrase Analysis SUBROUTINE.

Add partial word list entry (along with all related sublist entries) and two blank lines of text to Sentence-Analysis-DataSet2.txt.

**Total\_Num\_Sublist** = Determine total number of sublist entries of partial word list entry.

**IF** **Total\_Num\_Sublist** > 25 **THEN** **Total\_Num\_Sublist** = 25.

**FOR** **Index** = 1 **TO** **Total\_Num\_Sublist** **Step** 1

**Fourth\_Word[Index]** = Determine the fourth word of **Index** sublist entry.

**Next** **Index**

**Remark:** Above logic stores the fourth word of each sublist entry.

**Max\_Num\_Repeats** = 0.

**FOR** **Index** = 1 **TO** **Total\_Num\_Sublist** **Step** 1

**WW2** = **Estimated\_Text\_Sentence**(**NPAW** – 2).

**WW3** = **Estimated\_Text\_Sentence**(**NPAW** – 1).

**WW4** = **Fourth\_Word[Index]**.

**Next\_Word** = **Estimated\_Text\_Sentence**(**NPAW** + 1).

**Partial\_Word\_List\_Entry** = **WW2** + **spec** + **WW3** + **spc** + **WW4** + **spc** + **Next\_Word**.

**Remark:** **spc** = one space character.

**GOTO** Matching Phrase Analysis SUBROUTINE.

**Repeats** = Determine the number of repeats in the first sublist entry.

**IF** **Repeats** > **Max\_Num\_Repeats** **THEN**

**Max\_Num\_Repeats** = **Repeats**.

**Third\_Word** = Determine the third word of first sublist entry.

**ENDIF**

**Next** **Index**

**Remark:** Above logic determines the third word in a sublist entry associated to the highest number of repeats.

**Estimated\_Text\_Sentence(NPAW)** = **Third\_Word**

**Remark:** The third word in the sublist entry associated to the highest number of repeats has the highest probability of being the correct word.

**GOTO** Phrase Analysis SUBROUTINE.

**~~Final Phrase Analysis SUBROUTINE~~**

**~~Word~~** ~~=~~ **~~Complete\_Text\_Sentence(NPAW)~~**~~.~~

**~~Word\_Length~~** ~~= calculate length (in characters) of~~ **~~Word~~**~~.~~

**~~IF~~****~~Word\_Length~~** ~~= 1~~ **~~THEN~~**

~~Add 1 to~~ **~~Total\_Num\_Written\_Chars.~~**

**~~GOTO~~** ~~Complete Estimated Text Sentence SUBROUTINE.~~

**~~ENDIF~~**

**~~IF~~****~~Word\_Length~~** ~~= 2~~ **~~THEN~~**

~~Add 2.5 to~~ **~~Total\_Num\_Written\_Chars.~~**

**~~GOTO~~** ~~Complete Estimated Text Sentence SUBROUTINE.~~

**~~ENDIF~~**

**~~Remark:~~****~~PW1~~**~~,~~ **~~PW2~~**~~,~~ **~~PW3~~**~~, or~~ **~~PW4~~** ~~stand for partial word 1, 2, 3, or 4.~~

**~~WW1~~** ~~=~~ **~~Estimated\_Text\_Sentence~~**~~(~~**~~NPAW~~** ~~– 3).~~

**~~WW2~~** ~~=~~ **~~Estimated\_Text\_Sentence~~**~~(~~**~~NPAW~~** ~~– 2).~~

**~~WW3~~** ~~=~~ **~~Estimated\_Text\_Sentence~~**~~(~~**~~NPAW~~** ~~– 1).~~

**~~PW4~~** ~~= Determine text value of first 3 characters of~~ **~~Word~~**~~.~~

**~~Partial\_Word\_List\_Entry~~** ~~=~~ **~~WW1~~** ~~+~~ **~~spc~~** ~~+~~ **~~WW2~~** ~~+~~ **~~spc~~** ~~+~~ **~~WW3~~** ~~+~~ **~~spc~~** ~~+~~ **~~PW4~~**~~.~~

**~~Remark:~~****~~spc~~** ~~= one space character.~~

~~Add 3 to~~ **~~Total\_Num\_Written\_Chars~~**~~.~~

**~~Total\_Num\_Partial\_Characters~~** ~~= 3.~~

**~~GOTO~~** ~~Matching Phrase Analysis SUBROUTINE.~~

~~Add partial word list entry (along with all related sublist entries) and two blank lines of text to Sentence-Analysis-DataSet2.txt.~~

**~~Estimated\_Text\_Sentence(NPAW)~~** ~~= Determine the fourth word of first sublist entry.~~

**~~GOTO~~** ~~Complete Estimated Text Sentence SUBROUTINE.~~

**Matching Phrase Analysis SUBROUTINE**

Erase memory location used to store partial word list entry and all related sublist entries.

Add **Partial\_Word\_List\_Entry** to memory location.

Perform a search in coca-samples-text.txt for all whole 4-word phrases found within boundary of any given sentence that matches **Partial\_Word\_List\_Entry**.

**Remark:** The first 3 whole words in **Partial\_Word\_List\_Entry** must match the first 3 whole words in whole 4-word phrase and the fourth partial word in **Partial\_Word\_List\_Entry** must match first **Total\_Num\_Partial\_Characters** characters of the fourth word in whole 4-word phrase.

**IF** any matching whole 4-word phrase does **NOT** match anyone of the sublist entries **THEN**

Add matching whole 4-word phrase as numbered sublist entry (related to partial word list entry) along with 0 **repeats**.

**ENDIF**

**IF** any matching whole 4-word phrase matches one of the sublist entries **THEN**

Add one to value of **repeats** for sublist entry that matches whole 4-word phrase.

**ENDIF**

Once search is finished - re-arrange numbered sublist entries from highest to lowest repetitions.

**Complete Estimated Text Sentence SUBROUTINE**

**FOR** **Index** = 1 **TO** **Total\_Num\_Words\_RCS** **STEP** 1

**CTS** (Complete Text Sentence) = **CTS** + **Complete\_Text\_Sentence(Index)**.

**IF** **Index** < **Total\_Num\_Words\_RCS** **THEN** **CTS** = **CTS** + **spc**

**ETS** (Estimated Text Sentence) = **ETS** + **Estimated\_Text\_Sentence(Index)**.

**IF** **Index** < **Total\_Num\_Words\_RCS** **THEN** **ETS** = **ETS** + **spc**

**IF (Index > 3) OR (Index < Total\_Num\_Words\_RCS) THEN**

**IF** **Estimate\_Text\_Sentence(Index)** = **Complete\_Text\_Sentence(Index)** **THEN**

Add one to **Num\_Correct\_Words**.

**ELSE**

Add one to **Num\_Wrong\_Words**.

**ENDIF**

**ENDIF**

**NEXT** **Index**

**Total\_Num\_Correct\_Words** = **Total\_Num\_Correct\_Words** + **Num\_Correct\_Words**.

**Total\_Num\_Wrong\_Words** = **Total\_Num\_Wrong\_Words** + **Num\_Wrong\_Words**.

**IF** **Num\_Sentences\_Analyzed** = 1 **THEN**

Add, “This is a list of complete text sentences followed by estimated text sentences” to Complete-Estimated-Text-Sentence.txt.

Add blank line of text to Complete-Estimated-Text-Sentence.txt.

**ENDIF**

Add the following five items to Complete-Estimated-Text-Sentence.txt located in directory: C:\Text Analysis\Sentence Output Files\.

**CTS**, blank line of text, **ETS**, and blank line of text.

**GOTO** Sentence Analysis SUBROUTINE.

**Final Sentence Analysis SUBROUTINE**

When the word, “Add” is used – add the sentence/variable to Final-Sentence-Analysis-Part2.txt (located in C:\Text Analysis\Sentence Output Files\).

Add “Total number of sentences analyzed = “, **Total\_Num\_Sentences.**

Add blank line of text.

Add “Total number of words analyzed = “, **Total\_Num\_Words\_Analyzed**.

Add blank line of text.

**Average\_Sentence\_Length** = **(Total\_Num\_Words\_Analyzed + (Total\_Num\_Sentences X 4)) / Total\_Num\_Sentences).**

**Remark:** Round **Average\_Sentence\_Length** to 2 decimal places – example, 10.25, 14.50.

Add “Average length (in words) of each sentence = “, **Average\_Sentence\_Length**.

Add blank line of text.

**Average\_Word\_Length** = **Total\_Num\_CTS\_Chars / (Total\_Num\_Words\_Analyzed + (Total\_Num\_Sentences X 4)).**

**Remark:** Round **Average\_Word\_Length** to 2 decimal places – example, 10.25, 14.50.

Add “Average length (in characters) per word = “, **Average\_Word\_Length**.

Add blank line of text.

Add “Analysis completed in “, **HOURS**,” hours ”, **MINUTES**,” minutes, **SECONDS**,” seconds.”.

Add blank line of text.

**Time\_Analysis\_Seconds** = (**HOURS** X 3600) + (**MINUTES** X 60) + **SECONDS**.

**Num\_Words\_Analyzed\_Second** = **Total\_Num\_Words\_Analyzed** / **Time\_Analysis\_Seconds**.

**Num\_Words\_Analyzed\_Minute** = **Num\_Words\_Analyzed\_Second** X 60.

**Remark:** Round **Num\_Words\_Analyzed\_Minute** to 2 decimal places – example, 10.25, 14.50.

Add “Number of words analyzed per minute = “, **Num\_Words\_Analyzed\_Minute**.

Add blank line of text.

Add “The total number of texted or typed characters = “, **Total\_Num\_Inputted\_Chars**.

~~Add blank line of text.~~

~~Add “The total number of complete text sentence characters = “,~~ **~~Total\_Num\_CTS\_Chars~~**~~.~~

**TNWC** = **Total\_Num\_Inputted\_Chars.**

**Increase\_Input\_Rate** = **((Total\_Num\_CTS\_Chars – TNWC) / TNWC)** X 100.

Add “Increase in input rate of texting or typing = “, **Increase\_Input\_Rate**, “%” to Final-Sentence-Analysis.txt (located in C:\Text Analysis\Sentence Output Files\).

Add blank line of text.

**Total\_Num\_Words** = **Total\_Num\_Correct\_Words** + **Total\_Num\_Wrong\_Words**.

**Estimated\_Sentence\_Accuracy** = (**Total\_Num\_Correct\_Words** / **Total\_Num\_Words**) X 100.

Add “Accuracy rate of estimated text sentence = “,**Estimated\_Sentence\_Accuracy**,“%”.

Add blank line of text.