Number to MKS$ And MKS$ to Number.exe

Use *Number to MKS$ And MKS$ to Number.exe* to hide the necessary information within an encrypted file so that it can be decrypted.

(1) Create a random "junk" bytes file using all of the ASCII Characters. My program, *Random File Maker.exe* is useful for this. Only disclose the file **Length** to authorized receivers. Use a subtle way to disclose this **Length** to the receivers – some imagination is required. **(=Top.dat)**

(2) Using *Number to MKS$*, create a 4-byte file that will convert back to the **KEY** necessary to decrypt the "good" data bytes. **(=Key.dat)**

(3) Using *Number to MKS$*, create a 4-byte file that will convert back to the **LENGTH** of the "good" data bytes. **(=Length.dat)**

(4) The "good" data bytes that were *zipped* in order to preserve the name(s) and distinct file(s) of the content, *and then* encrypted. **(=Encrypted.dat)**

(5) Create another random "junk" bytes file using all of the ASCII Characters as done in (1), but of a different length. The length does not have to be disclosed to the receivers. **(=Bottom.dat)**

(6) Concatenate **(1) + (2) + (3) + (4) + (5)** in that order. This can be done with my program, *Concatenate Two Files.exe*, this way:

Top.dat + Key.dat = 1.dat

1.dat + Length.dat = 2.dat

2.dat + Encrypted.dat = 3.dat

3.dat + Bottom.dat = 4.dat

(7) Finally, encrypt **4.dat** using as the key, the **Length** of **Top.dat** file as the encryption key. **(=Finished.dat)**

NOTE: In addition to disclosing the **Length** of **Top.dat** to **ONLY** authorized receivers, also **ONLY** disclose the **method of encryption** to these authorized receivers.

To Decrypt

(A) Decrypt **Finished.dat** using the **Length** of the original **Top.dat** file as the decryption key. This key is **NOT** the same as the KEY created in (2) and found in (B). **(=4.dat)**

(B) Using *MKS$ to Number*, go into **4.dat** at the offset (**Length of Top.dat + 1**) to get the **KEY** necessary to decrypt the "good" data bytes.

(C) Using *MKS$ to Number*, go into **4.dat** at the offset (**Length of Top.dat + 5**) to get the **LENGTH** of the "good" data bytes.

(D) Using *Extract Bytes From Any File.exe*, extract the "good" data bytes at the offset (**Length of Top.dat + 9**) with a **LENGTH** that was created in (3) and found in (C). **(=Encrypted.dat)**

(E) Decrypt **Encrypted.dat** using the **KEY** that was created in (2) and found in (B) **(=AnyName.zip)** Notice that the name of the zip file does not matter.

(F) Unzip **AnyName.zip** to retrieve the file(s).